



**PROYECTO FIN DE CARRERA
PLAN 2000**

E.U.I.T. TELECOMUNICACIÓN

TEMA: Eficiencia energética y Medio ambiente.

TÍTULO: Modelos predictivos y preventivos
para una Administración Pública sostenible.

AUTOR: José Manuel Estévez Jorge

TUTOR: Sara Lana Serrano

Vº Bº.

DEPARTAMENTO: DIATEL

Miembros del Tribunal Calificador:

PRESIDENTE:

VOCAL:

VOCAL SECRETARIO:

DIRECTOR:

Fecha de lectura:

Calificación:

El Secretario,

RESUMEN DEL PROYECTO:

Con la finalidad de **ayudar a la creación y desarrollo de modelos de predicción y simulación** que permitan al ciudadano/administraciones publicas gestionar el consumo energético de forma más eficiente y respetuosa con el medio ambiente, se ha implementado un **sistema de gestión de datos de indicadores energéticos**. Se ha pretendido realizar una **aplicación** lo más **abierta** posible, tanto desde el punto de vista de la funcionalidad, permitiendo la **definición del propio indicador** a través del sistema, como desde el punto de vista de la implementación, **usando únicamente código abierto** para el desarrollo de la misma.



E.U.I.T. TELECOMUNICACIÓN

**PROYECTO FIN DE CARRERA
PLAN 2000**

TEMA: Eficiencia energética y Medio ambiente.

TÍTULO: Modelos predictivos y preventivos
para una Administración Pública sostenible.

AUTOR: José Manuel Estévez Jorge

TUTOR: Sara Lana Serrano

Vº Bº.

DEPARTAMENTO: DIATEL

Miembros del Tribunal Calificador:

PRESIDENTE: Pedro Garcia del Pino

VOCAL: Sara Lana Serrano

VOCAL SECRETARIO: José Luis López Presa

DIRECTOR:

Fecha de lectura: 27 de septiembre de 2013

RESUMEN DEL PROYECTO:

Con la finalidad de **ayudar a la creación y desarrollo de modelos de predicción y simulación** que permitan al ciudadano/administraciones publicas gestionar el consumo energético de forma más eficiente y respetuosa con el medio ambiente, se ha implementado un **sistema de gestión de datos de indicadores energéticos**.

En **2007** la UE creó una directiva conocida como "**20/20/20**" en la que se la Unión Europea se compromete a ahorrar un 20% del consumo anual de energía primaria desde esa fecha a 2020.

En **2009** la Comisión Europea ha llegado a la conclusión de que con las medidas propuestas en dicha directiva **no se podría alcanzar el objetivo** de reducción del 20% del consumo energético previsto para el 2020, quedándose en menos de la mitad.

Para dar un nuevo **impulso** a la eficiencia energética se redacta una propuesta de directiva: **2011/0172(COD)**. En esta directiva se obliga a los estados miembros a potenciar y ampliar la información estadística agregada sobre sus clientes finales (los perfiles de carga, la segmentación de los clientes, su ubicación geográfica, etc).

La Unión Europea plantea **que incrementar el volumen y la accesibilidad de los datos** de consumo energético, ayudará de forma significativa a alcanzar los objetivos.

En este marco, parece lógico afirmar que un **banco de datos de indicadores energéticos** universalmente accesible puede contribuir de un modo efectivo al aumento de la **eficiencia energética**.

Como aplicativo de este PFC se ha **desarrollado una aplicación** que permite la definición y almacenamiento de indicadores energéticos, en la que los diferentes sistemas, propietarios o abiertos, pueden volcar y extraer datos de una forma **poco costosa**.

Se ha pretendido realizar una aplicación lo más **abierta** posible, tanto desde el punto de vista de la funcionalidad, permitiendo la **definición** del propio **indicador** a través del sistema, como desde el punto de vista de la implementación, usando únicamente **código** abierto para el desarrollo de la misma.



E.U.I.T. TELECOMUNICACIÓN

**PROYECTO FIN DE CARRERA
PLAN 2000**

TEMA: Eficiencia energética y Medio ambiente.

TÍTULO: Modelos predictivos y preventivos
para una Administración Pública sostenible.

AUTOR: José Manuel Estévez Jorge

TUTOR: Sara Lana Serrano

Vº Bº.

DEPARTAMENTO: DIATEL

Miembros del Tribunal Calificador:

PRESIDENTE: Pedro Garcia del Pino

VOCAL: Sara Lana Serrano

VOCAL SECRETARIO: José Luis López Presa

DIRECTOR:

Fecha de lectura: 27 de septiembre de 2013

ABSTRACT:

In order to assist in the creation and development of forecasting and simulation models that enable citizens / public authorities manage energy consumption more efficient and environmentally friendly, we have implemented a data management system of energy indicators . In 2007 the EU created a policy known as " 20/20/20 " in which the European Union is committed to saving 20 % of the annual primary energy consumption from that date to 2020 .

In 2009 the European Commission has concluded that the measures proposed

in the directive could not achieve the goal of 20% reduction in energy consumption expected for 2020 , staying in less than half. To give new impetus to energy efficiency is drawn up a draft directive : 2011/0172 (COD) . This directive obliges member states to strengthen and expand aggregate statistical information on their final customers (load profiles , customer segmentation , geographic location, etc.) . The European Union argues that increasing the volume and accessibility of energy data , will significantly help to achieve the objectives . In this context , it seems logical to say that a database of universally accessible energy indicators can contribute in an effective way to increase energy efficiency.

As of this PFC application has developed an application that allows the definition and storage of energy indicators , in which different systems, proprietary or open, can tip and extract data from an inexpensive way . We have tried to make an application as open as possible , both from the point of view of functionality , allowing the definition of the indicator itself through the system , and from the point of view of implementation, using only open source development thereof.

Modelos predictivos y preventivos para una Administración Pública sostenible.

Modelos predictivos y preventivos para una Administración Pública sostenible.

Índice de contenidos

1	AGRADECIMIENTOS.....	8
2	INTRODUCCIÓN	9
2.1	MEDIR E INFORMAR: UN GRAN PASO PARA ALCANZAR LOS OBJETIVOS	11
2.2	CONTADORES INTELIGENTES	13
2.3	SISTEMAS DE MEDICIÓN INTELIGENTE.....	15
2.3.1	<i>ENEL/ENDESA.....</i>	<i>15</i>
2.3.2	<i>YELLO STROM.....</i>	<i>16</i>
2.3.3	<i>SIEMENS.....</i>	<i>17</i>
3	BASE TEÓRICA	18
4	DESCRIPCIÓN EXPERIMENTAL	20
4.1	OBJETIVO: DESARROLLO DE UN SISTEMA DE GESTIÓN DE DATOS DE INDICADORES ENERGÉTICOS 20	
4.2	CAPTURA DE REQUISITOS	21
4.3	ESPECIFICACIÓN DE REQUISITOS	22
4.3.1	<i>Introducción</i>	<i>22</i>
4.3.1.1	<i>Propósito</i>	<i>22</i>
4.3.1.2	<i>Ámbito del sistema.....</i>	<i>22</i>
4.3.1.3	<i>Definiciones, acrónimos y abreviaturas</i>	<i>22</i>
4.3.2	<i>Descripción general.....</i>	<i>24</i>
4.3.2.1	<i>Perspectiva del producto.....</i>	<i>24</i>
4.3.2.2	<i>Funciones del producto.....</i>	<i>24</i>
4.3.2.3	<i>Características de los usuarios</i>	<i>26</i>
4.3.2.4	<i>Restricciones.....</i>	<i>26</i>
4.3.2.5	<i>Supuestos y dependencias</i>	<i>26</i>
4.3.2.6	<i>Requisitos Futuros.....</i>	<i>27</i>
4.3.3	<i>Requisitos específicos.....</i>	<i>27</i>
4.3.3.1	<i>Interfaces externos</i>	<i>27</i>
4.3.3.2	<i>Requerimientos funcionales.....</i>	<i>28</i>
4.3.3.3	<i>Rendimiento</i>	<i>32</i>
4.3.3.4	<i>Atributos</i>	<i>32</i>
4.4	DISEÑO	33
4.4.1	<i>Modelo de datos.....</i>	<i>35</i>
4.4.1.1	<i>Subsistema Identidad Única.....</i>	<i>35</i>
4.4.1.2	<i>Subsistema Localizaciones.....</i>	<i>35</i>

4.4.1.3	Subsistema Árbol Jerárquico.....	36
4.4.1.4	Subsistema Clasificación	37
4.4.1.5	Subsistema Gestión de Indicadores.....	38
4.4.1.6	Atributos de la definición del Indicador	39
4.4.1.7	Atributos del valor de un indicador	40
4.4.2	<i>Interfaces</i>	41
4.4.2.1	XMLRPC	41
4.4.2.2	REST.....	42
4.4.2.3	PHP.....	43
4.5	CASOS DE USO.....	43
4.5.1	<i>Indicadores energéticos definidos por E2TIC</i>	43
4.5.2	<i>Seguimiento del consumo del sistema de luminarias en una administración local</i> ...	44
4.6	VALIDACIÓN	44
5	PLANOS	44
6	PRESUPUESTO	45
7	MANUAL DE USUARIO	46
7.1	GUÍA DE INSTALACIÓN	46
7.2	GUÍA DE ADMINISTRACIÓN.....	47
7.3	GUÍA DE USUARIO	48
7.3.1	<i>Herramientas de ayuda al desarrollo</i>	49
7.3.1.1	Herramienta de pruebas de los servicios.....	49
7.3.2	<i>Herramientas de manipulación de datos</i>	50
7.3.2.1	Tablas CRUD	50
7.3.2.2	Árboles de selección y clasificación	51
8	CONCLUSIONES	53
9	ANEXO I: INDICADORES DE EFICIENCIA ENERGÉTICA.....	55
9.1	CONSUMO ENERGÉTICO PER CÁPITA	56
9.2	PROPORCIÓN DE CONSUMO DE ENERGÍAS RENOVABLES	57
9.3	VEHÍCULOS ELÉCTRICOS	58
9.4	CIUDADES INTELIGENTES.....	59
9.5	EDIFICIOS INTELIGENTES	60
9.6	LOGÍSTICA INTELIGENTE	61
10	ANEXO II: DETALLES DE IMPLEMENTACIÓN.....	62
10.1	SELECCIÓN DEL FRAMEWORK: DRUPAL (CMS)	62
10.1.1	<i>Drupal, módulos del core</i>	63
10.2	ESTRUCTURA DE FICHEROS	65

10.3	FUNCIONALIDAD TRANSVERSAL	65
10.3.1	<i>devel - Ayuda al desarrollo</i>	66
10.3.2	<i>multienvironment - entorno local y de producción</i>	67
10.3.3	<i>allinone - Todo en un módulo</i>	67
10.3.4	<i>accessbylevel - acceso a nodos por nivel</i>	68
10.3.5	<i>scope - encapsulamiento de contenidos</i>	70
10.3.6	<i>customblock - bloques por scope y tema</i>	72
10.3.7	<i>plugin - funcionalidad encapsulada y opcional</i>	73
10.3.8	<i>profiling - diagnóstico y optimización del sistema</i>	74
10.4	LIBRERÍAS.....	75
10.4.1	<i>JDT - jQuery Datatables</i>	76
10.4.1.1	Plugin jQuery Datatables (v1.9.4).....	77
10.4.1.2	Plugin jQuery jeditable (v.1.7.2)	78
10.4.1.3	Plugin jquery-ui (v1.9.2)	78
10.4.1.4	Plugin jQuery validation (v1.11.0).....	78
10.4.1.5	Plugin jQuery dataTable.editable (v2.3.3).....	78
10.4.1.6	Plugin jQuery dataTable.columnFilter (v1.5.0)	79
10.4.1.7	Plugin jQuery KeyTable (v1.1.7).....	79
10.4.1.8	Plugin jQuery alerts (v1.1).....	79
10.4.1.9	Plugin jQuery jloadable (toolkit).....	79
10.4.2	<i>JTR - jQuery jsTree</i>	80
10.4.3	<i>RPC_XML_a0m - Interfaz xmlrpc</i>	81
10.4.4	<i>OWN - Repositorio común de objetos</i>	82
10.4.5	<i>CLIPHP - Interfaz php con servicios</i>	83
10.5	FUNCIONALIDAD ESPECÍFICA	83
10.5.1	<i>Plugins</i>	83
10.5.2	<i>Scopes - subsistemas</i>	85
10.5.3	<i>Detalle de los interfaces para el Servicio svcid</i>	85
REFERENCIAS.....		88

Índice de figuras

Figura 1 Sistema Central de Monitoreo y Control Remoto Vía Internet. ACEE consultores.....	11
Figura 2 Proceso de estimación, evaluación, medición y seguimiento de oportunidades de eficiencia energética [3].....	12
Figura 3 Esquema de funcionamiento de un sistema de monitorización [4].....	13
Figura 4 Esquema del sistema de medición inteligente propuesto por ENEL.....	15
Figura 5 Esquema del sistema de medición inteligente propuesto por YELLOW STROM.....	16
Figura 6 Esquema del sistema para el control energético de plantas HVAC de SIEMENS.....	17
Figura 7 Árbol jerárquico. Modelo de lista adyacente.....	18
Figura 8 Árbol jerárquico. Modelo de conjuntos anidados.....	19
Figura 9 Banco de Datos de Indicadores Energéticos	20
Figura 10 Requerimientos para la definición y valores de un indicador energético...	25
Figura 11 Esquema del Sistema de Gestión de Datos de Indicadores Energéticos..	34
Figura 12 Esquema de datos del subsistema 'identidad única'	35
Figura 13 Esquema de datos del subsistema 'localizaciones'.....	35
Figura 14 Esquema de datos del subsistema 'árbol jerárquico'	36
Figura 15 Esquema de datos del subsistema 'clasificación'.....	37
Figura 16 Esquema de datos del subsistema 'gestión de indicadores'	38
Figura 17 Ejemplo de petición XMLRPC.....	42

Figura 18 Ejemplo de respuesta XMLRPC.....	42
Figura 19 Página inicial de la aplicación.....	48
Figura 20 Página de herramientas de ayuda al desarrollo.....	49
Figura 21 Herramienta de pruebas de los servicios.....	50
Figura 22 Tabla CRUD	51
Figura 23 Árboles de selección y clasificación	52
Figura 24 Uso de las diferentes fuentes de energía en España [21].....	56
Figura 25 Ejemplo solución Smart City por MetricStream [23].....	60
Figura 26 Cambios en el esquema para la funcionalidad <i>accessbylevel</i>	69
Figura 27 Cambios en el esquema para la funcionalidad <i>scope</i>	71
Figura 28 Esquema de datos para la funcionalidad <i>themeblock</i>	73

1 Agradecimientos

Quiero agradecer a Sara Lana, mi tutor de proyecto, su amabilidad e inestimable ayuda en la realización del mismo.

A mis familiares y amigos, por su apoyo y el cariño recibido, ellos han sufrido mi falta de atención durante la realización del mismo. En especial a mi mujer, Pachi, y a mis padres, Aurora y José Antonio.

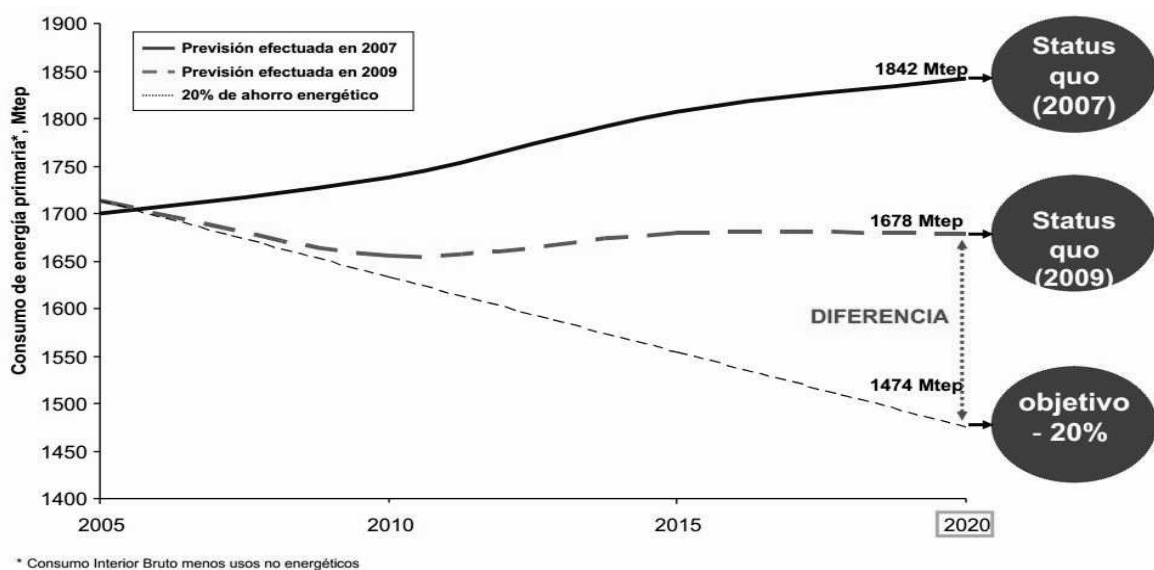
2 Introducción

A finales de 2006, la Unión Europea (UE) se comprometió a ahorrar un 20% del consumo anual de energía primaria desde esa fecha a 2020.

En 2007 la UE creó una directiva conocida como "20/20/20" en la que se establecieron los siguientes compromisos:

- **Reducir en un 20% las emisiones de gases** de efecto invernadero con respecto a los niveles de 1990.
- **Ahorrar el 20% del consumo de energía** mediante una mayor eficiencia energética
- Elevar hasta el **20% el peso de las energías renovables**

La Comisión Europea ha llegado a la conclusión de que con las medidas propuestas en dicha directiva no se podría alcanzar el objetivo de reducción del 20% del consumo energético previsto para el 2020, quedándose en menos de la mitad. En la siguiente figura se observa que la previsión realizada en 2009 se queda a 204 Mtep¹ del objetivo. [1]



¹ Mtep: 1 millón de toneladas equivalentes de petróleo.

Es por ello que se ha redactado una propuesta de directiva que intenta dar un nuevo impulso a la eficiencia energética: **2011/0172(COD)**.

En esta propuesta se hace especial énfasis en temas relacionados con **auditorías energéticas** y con dar al consumidor final **mayor información sobre sus hábitos de consumo**.

“Para aprovechar el potencial de ahorro energético de determinados segmentos de mercado en los que no suelen ofrecerse auditorías energéticas de forma comercial (como los hogares o las pequeñas y medianas empresas), los Estados miembros deben asegurar la posibilidad de efectuar auditorías energéticas. Estas tienen que ser obligatorias y periódicas para las grandes empresas ya que los ahorros de energía obtenidos pueden ser significativos.”

En concreto el punto 6 del artículo 6 dice:

Los Estados miembros publicarán los ahorros energéticos obtenidos por cada parte obligada, así como información sobre la tendencia anual de los ahorros energéticos dentro del régimen. A los efectos de publicar y verificar los ahorros energéticos conseguidos, los Estados miembros requerirán a las partes obligadas para que les presenten, al menos, los datos siguientes:

- a) los ahorros energéticos conseguidos;*
- b) información estadística agregada sobre sus clientes finales (señalando cambios significativos en la información anteriormente presentada); e*
- c) información actual sobre el consumo de los clientes finales, incluidos, en su caso, los perfiles de carga, la segmentación de los clientes y su ubicación geográfica, preservando, al mismo tiempo, la integridad y confidencialidad de la información privada o comercialmente sensible, en cumplimiento de la legislación aplicable de la Unión Europea.*

2.1 Medir e informar: un gran paso para alcanzar los objetivos

La definición de **auditoría energética** que realiza la directiva es la siguiente:

*“procedimiento sistemático para obtener conocimientos adecuados del perfil de consumo de energía de un edificio o grupo de edificios, de una instalación u operación industrial o comercial, o de un servicio privado o público, así como para determinar y **cuantificar** las posibilidades de ahorro de energía a un coste eficiente e **informar** al respecto”*

Parece evidente que la obtención, manejo y difusión de los **indicadores** energéticos constituyen un aspecto muy relevante en el entorno de la **eficiencia** energética.

Tanto desde el sector privado como el público se hace énfasis en este aspecto. Por ejemplo, en la página web de ACEE Consultores [2], empresa especializada en sistemas de monitorización energética, se destaca lo siguiente:

“.. contar con un sistema de medición permanente lleva a los usuarios tarde o temprano a lograr mejoras en el sistema eléctrico ...”

“ ...Información Estadística de la IEEE menciona que simplemente por instalar un sistema de monitorización de energía eléctrica, se tiene ahorros del 1% ...”

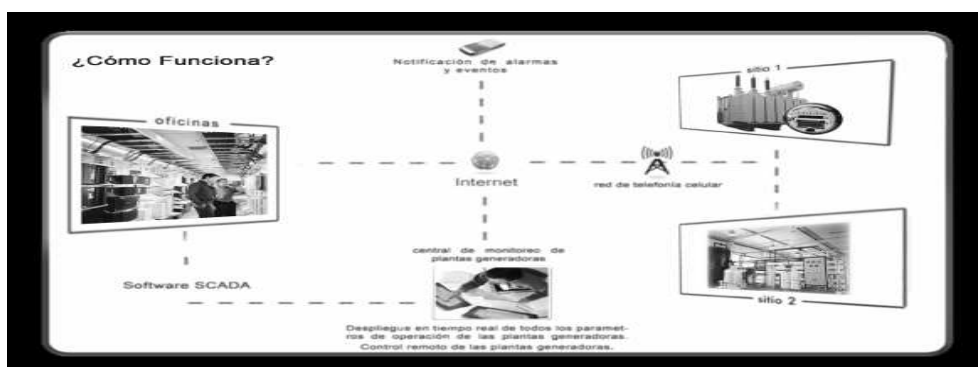


Figura 1 Sistema Central de Monitoreo y Control Remoto Vía Internet. ACEE consultores.

La administración australiana ha publicado una GUÍA DE MEDICIÓN DEL AHORRO DE ENERGÍA [3], que trata de cómo estimar, medir, evaluar y hacer un seguimiento oportunidades de eficiencia energética. En su capítulo 2.3, que trata acerca de cómo desarrollar un sistema de medición energética, se puede leer:

“Sin datos suficientes, el consumo de energía no se puede medir, el ahorro de energía no se puede pronosticar, oportunidades de energía no pueden ser evaluados, y el ahorro de energía no pueden ser seguidos ni reportados.”

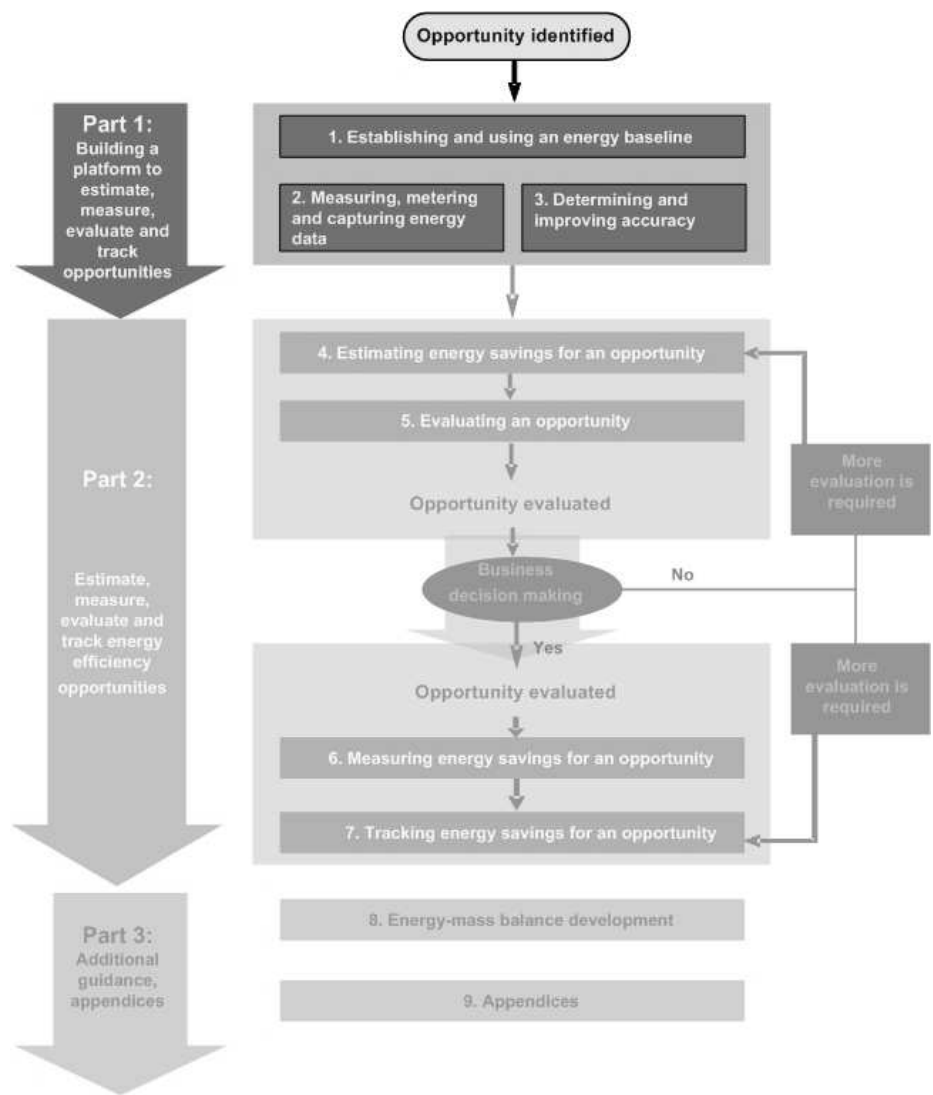


Figura 2 Proceso de estimación, evaluación, medición y seguimiento de oportunidades de eficiencia energética [3]

2.2 Contadores inteligentes



Los contadores o medidores inteligentes, **smart meters** en inglés, son unos dispositivos avanzados que, además de calcular el consumo, son capaces de transmitir esta información a través de alguna red a un centro de control.

Las actuales directivas comunitarias prevén que los contadores tradicionales actuales sean completamente reemplazados por inteligentes para finales del año 2022 como muy tarde y, atendiendo al objetivo 20/20/20, el 80% para el 2020.

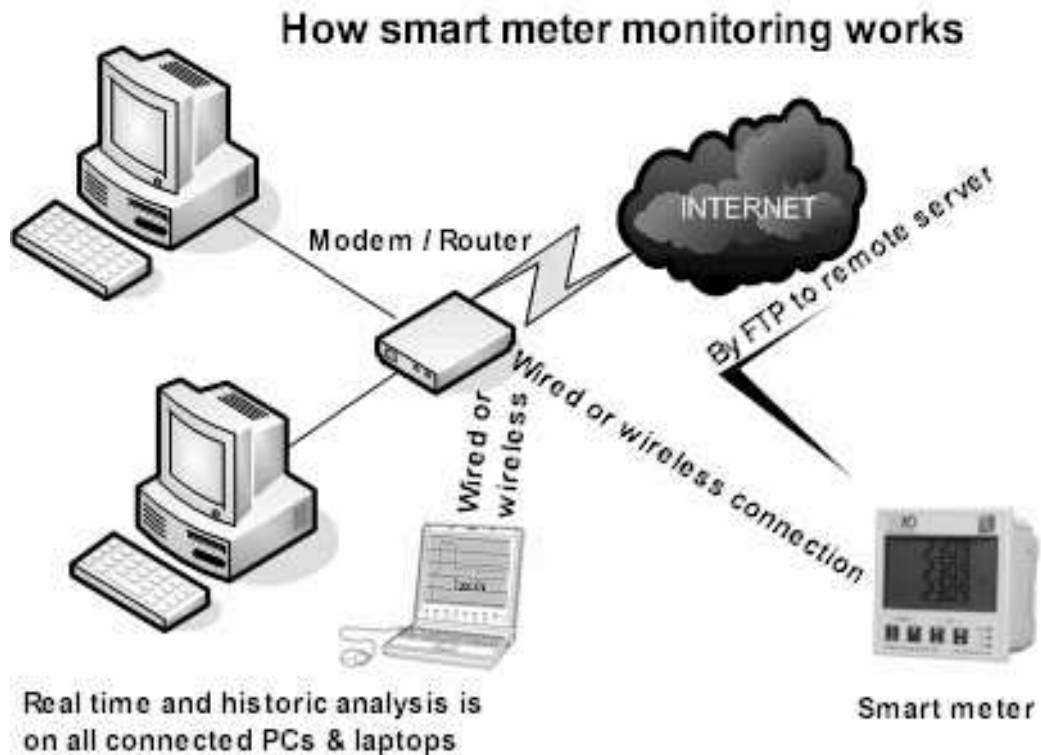


Figura 3 Esquema de funcionamiento de un sistema de monitorización [4]

La información suministrada por los contadores inteligentes puede ser usada no solo para fines de facturación sino también para otros relacionados con la eficiencia energética:

- **Conexión/desconexión** de dispositivos.
- Información en tiempo real de nuestro **consumo**.
- Información en tiempo real del **coste**.

El uso extendido de estos nuevos contadores inteligentes supondrá un gran impulso grande para alcanzar los objetivos de eficiencia energética marcados desde la unión europea [5]:

“La medición inteligente combinada con una información directa al consumidor se ha demostrado que aumenta la eficiencia energética en un 5-15%, y en algunos casos incluso hasta 20%”.

Existe un mandato de la Unión Europea, M/441 [6] para la creación de normas europeas que permitan la **interoperabilidad** de los medidores inteligente usados por los servicios públicos.

Con este objetivo se nació el proyecto **OPEN Meter**, que pretendía la adopción de estándares abiertos para equipos medidores inteligentes, proporcionando cobertura a los siguientes aspectos relevantes:

- marcos normativos
- funciones inteligentes de medición
- medios de comunicación
- protocolos
- formatos de datos
- ...

2.3 Sistemas de medición inteligente

Existen en el mercado diferentes sistemas de medición inteligente. A continuación se enumeran los más destacados a nivel europeo.

2.3.1 ENEL/ENDESA

Este sistema está desarrollado conjuntamente por Endesa y Enel.

El contador inteligente es el elemento principal del sistema. Con este sistema se encuentran desplegados más de **33 millones de contadores inteligentes** [7] en todo el mundo. Endesa está desarrollando un plan de sustitución de todos los contadores de los clientes con potencia contratada de hasta 15 kW.

El sistema de tele-gestión de Endesa utiliza la tecnología de comunicaciones llamada **Meters & More** [8]. Esta tecnología es abierta, interoperable, en proceso de estandarización a nivel europeo y ha sido puesta a disposición de todas las empresas del sector a través de una asociación sin ánimo de lucro.

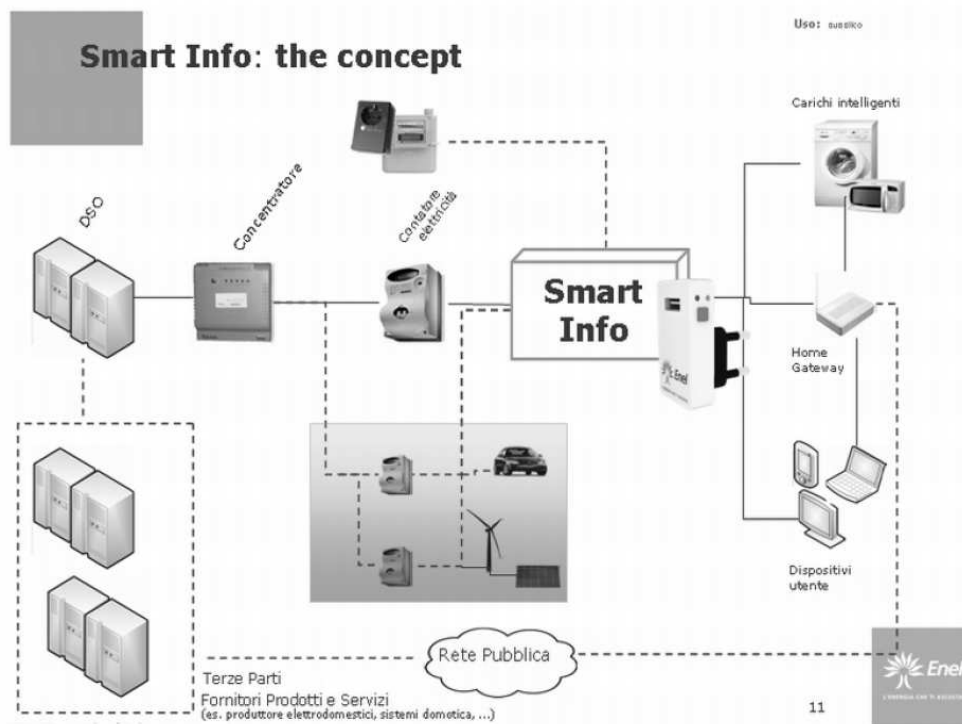


Figura 4 Esquema del sistema de medición inteligente propuesto por ENEL.

2.3.2 YELLO STROM

La compañía alemana Yello Strom es una filial de la tercera compañía energética más grande de Alemania: Energie Baden-Württemberg.

Yello Strom desarrolla y vende sus propios contadores inteligentes: el medidor Sparzähler ("medidor de ahorro"). Éste **gestiona el servicio de contador inteligente directamente a través de conexión de banda ancha de sus clientes** [9].

Es la primera compañía europea que ofrece a sus clientes acceso a la herramienta de gestión de energía **PowerMeter de Google** [10], y dispone de un prototipo de aplicación que permite *twittear* el consumo de energía de sus clientes.

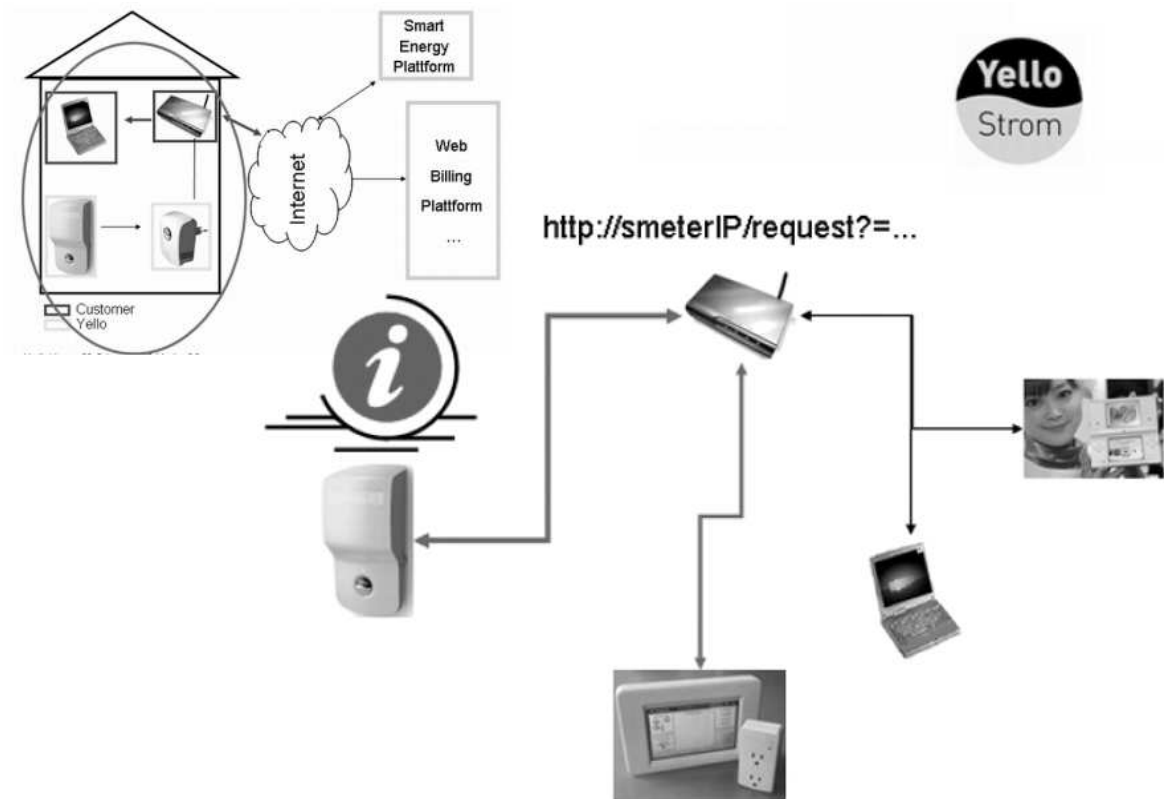


Figura 5 Esquema del sistema de medición inteligente propuesto por YELLOW STROM.

2.3.3 SIEMENS

La división “*Building Technologies*” de Siemens ha desarrollado la herramienta software “Indicador de energía”, que se utiliza para aumentar la eficiencia energética de las plantas HVAC. Se puede utilizar con todos los controladores de Siemens que tienen un bus de comunicaciones KNX o LPB.

Esta herramienta permite **controlar los dispositivos a través de Internet**. El usuario puede **comprobar** si los parámetros seleccionados son energéticamente eficientes y los operadores pueden recibir una notificación por correo electrónico cuando la planta no está se **utiliza la energía de manera eficiente**.

En el año fiscal 2011 los productos y soluciones de Siemens permitieron a sus clientes a reducir las emisiones de dióxido de carbono (CO₂) en cerca de 320 millones de toneladas [11].

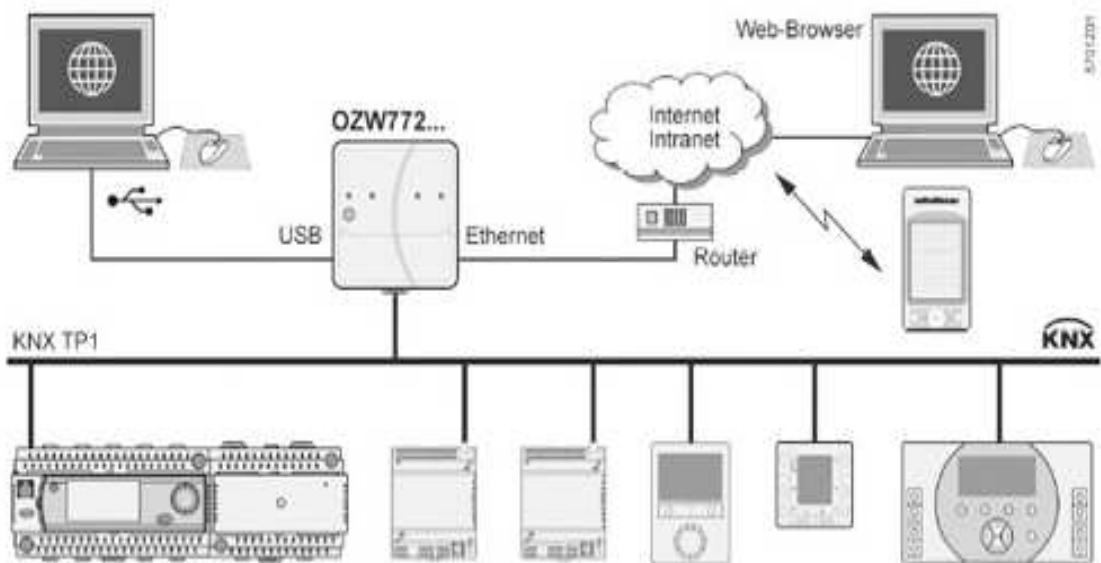


Figura 6 Esquema del sistema para el control energético de plantas HVAC de SIEMENS.

3 Base Teórica

El proyecto realizado ha sido eminentemente práctico, produciendo como resultado el prototipo de un sistema de gestión de datos sobre una plataforma web. Para la realización del mismo se han aplicado los conocimientos teóricos adquiridos en las diferentes asignaturas impartidas en la EUITT, especialmente los adquiridos en la Asignatura Tecnología Web.

Para optimizar la implementación de las estructuras de árboles jerárquicos, se ha profundizado en el conocimiento de las mismas. Se ha intentado hacer uso de diferentes modelos, intentando usar el más eficaz en cada caso [13]. A continuación se menciona los dos modelos usados y la principal característica de cada uno de ellos.

El modelo de **lista adyacente** es **rápido en las inserciones** ya que la única información que se necesita es la identidad del nodo padre.

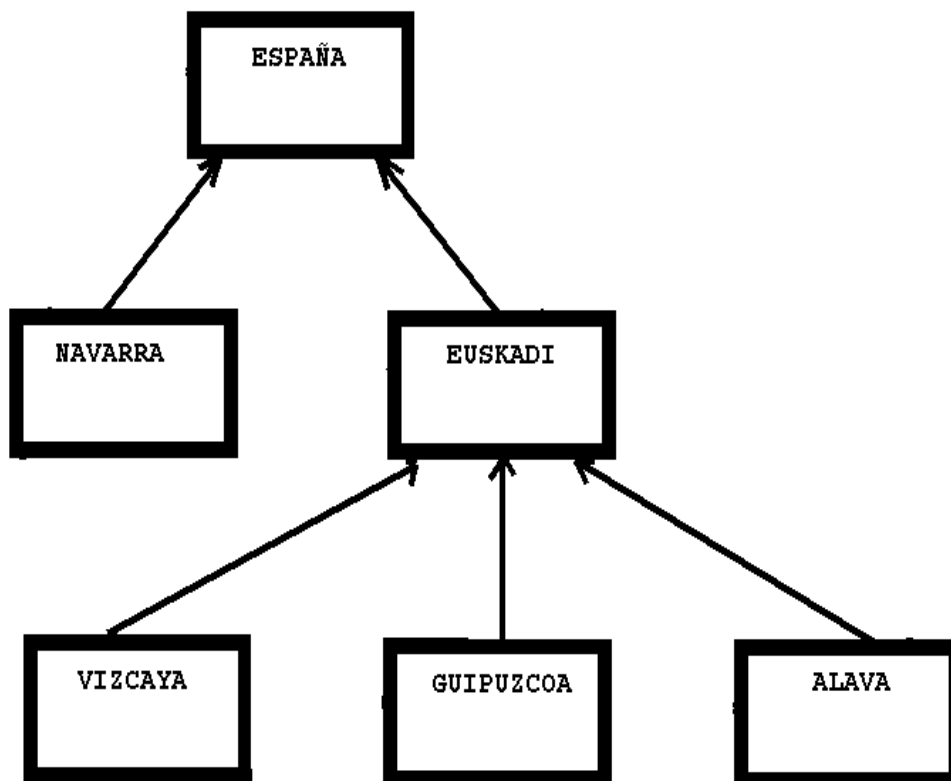


Figura 7 Árbol jerárquico. Modelo de lista adyacente

El modelo de **conjuntos anidados** es lento en las inserciones pero a cambio es **rápido a la hora de buscar caminos, encontrar nodos finales, o el conjunto de nodos subordinados**.

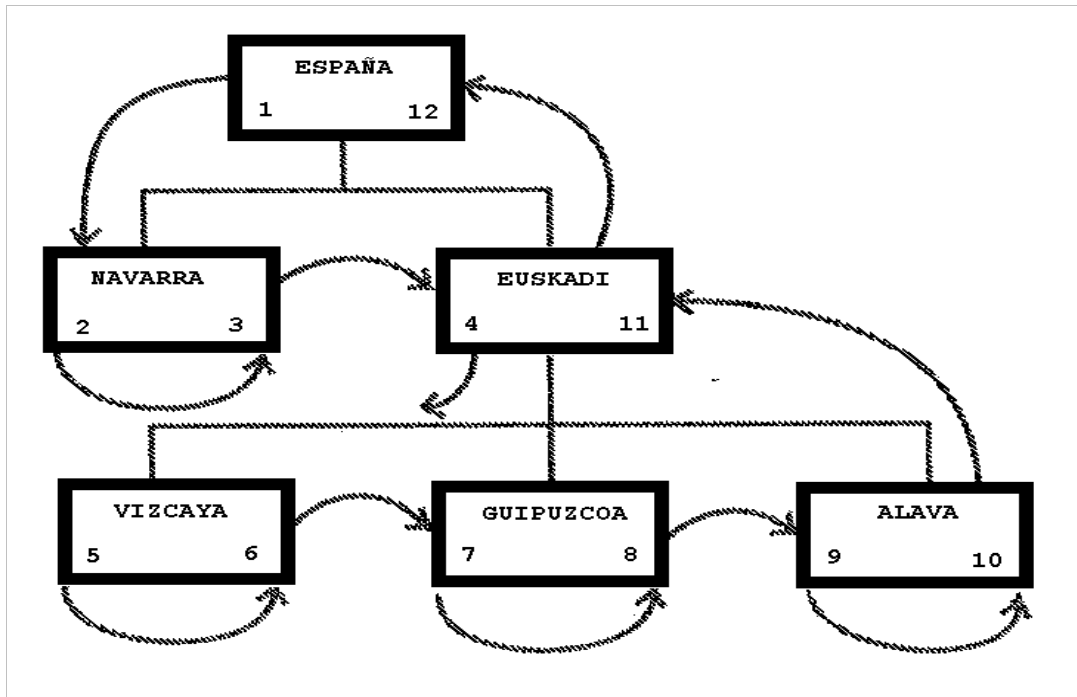


Figura 8 Árbol jerárquico. Modelo de conjuntos anidados

4 Descripción experimental

4.1 Objetivo: Desarrollo de un Sistema de Gestión de Datos de Indicadores Energéticos

En la actualidad todas las compañías distribuidoras de energía están trabajando, obligadas o no por las diferentes regulaciones o por la presión de los consumidores, en desplegar contadores inteligentes que proporcionen mayor información y control del consumo energético a sus clientes. La Unión Europea tiene una voluntad clara de que estos dispositivos operen bajo unas bases de interoperabilidad que potencie y aumente la eficiencia energética de las instalaciones.

Si el objetivo es incrementar el volumen y accesibilidad de los datos de consumo energético, parece lógico afirmar que **un banco de datos de indicadores energéticos universalmente accesible puede contribuir de un modo efectivo al aumento de la eficiencia energética.**

Como aplicativo de este PFC se ha desarrollado una aplicación que permite la **definición y almacenamiento de indicadores energéticos**, en la que los diferentes sistemas, propietarios o abiertos, pueden volcar y extraer datos de una forma poco costosa.

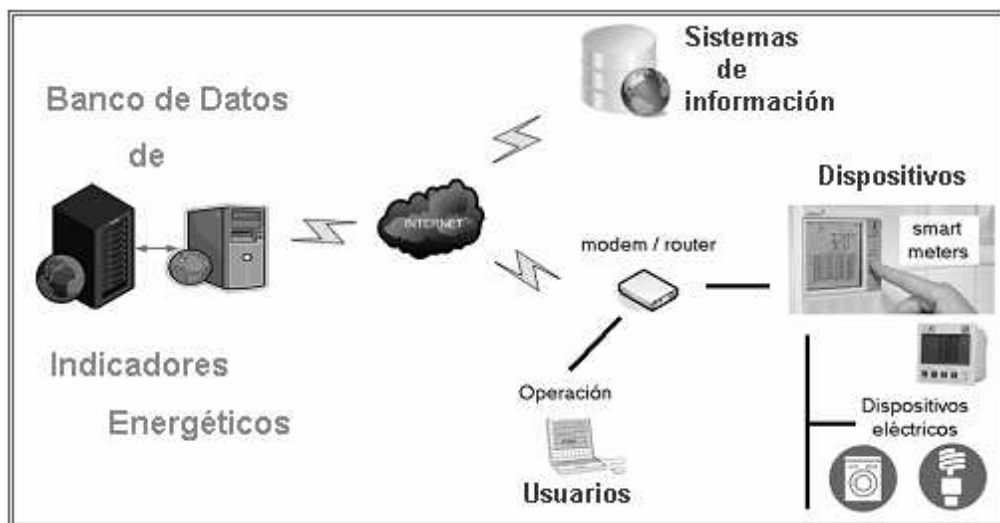


Figura 9 Banco de Datos de Indicadores Energéticos

4.2 Captura de requisitos

Con el objeto de cubrir todas las posibles necesidades en cuanto a la definición de indicadores que puedan ser definidos y almacenados en el banco de datos, se ha consultado diversas fuentes de datos, entre las que se destacan las siguientes:



Base de datos estadísticos energéticos de las Naciones Unidas [13].



Base de datos de consumo energético del Instituto Nacional de Estadística de España [14].



Base de datos de energía limpia en formatos abiertos, Reegle Data [15].

Del análisis de estas fuentes de datos y de los objetivos planteados por el tutor para esta aplicación, se han generado la especificación de requisitos [16] expuestos en el siguiente apartado.

4.3 Especificación de requisitos

4.3.1 Introducción

4.3.1.1 Propósito

El propósito de la especificación de requisitos es definir los requerimientos que debe tener la aplicación que se va a desarrollar y describir su funcionalidad.

4.3.1.2 Ámbito del sistema

La aplicación está orientada a ofrecer un conjunto de funcionalidades que ayuden a la caracterización, definición y almacenamiento de **cualquier indicador energético** que permita realizar un **uso más eficiente de la energía**.

En la aplicación existen dos partes muy diferenciadas: la **definición** de un indicador energético y el **almacenamiento/consulta** del indicador energético.

4.3.1.3 Definiciones, acrónimos y abreviaturas

Banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso [17].

Sistema de Gestión de Bases de Datos (SGBD) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos [17].

Servidor web: Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente, generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente [17].

Navegador o **navegador web**: es un software que permite el acceso a Internet, interpretando la información de archivos y sitios web para que éstos puedan ser leídos [17].

Interfaz: Parte del programa informático que permite el flujo de información entre varias aplicaciones o entre el propio programa y el usuario [18].

Interfaz REST: La Transferencia de Estado Representacional (Representational State Transfer) o REST es una técnica de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web. Describe cualquier interfaz web simple que utiliza XML y HTTP, sin las abstracciones adicionales de los protocolos basados en patrones de intercambio de mensajes como el protocolo de servicios web SOAP. REST es un protocolo cliente/servidor sin estado y sus sistemas se caracterizan por tener un conjunto de operaciones bien definidas [17].

Indicadores: parámetros de medición que integran generalmente más de un variable básica que caracteriza un evento, a través de formulaciones matemáticas sencillas, ampliando el significado de las variables que lo componen y permitiendo una más fácil comprensión de las causas, comportamiento y resultados de una actividad [19].

Localización: identificación de un punto o área de la superficie terrestre.

Tipo de Localización: división de la superficie terrestre o de parte de ella.

Referencia: en el contexto de un indicador, es el objeto al cual se refiere el indicador. Por ejemplo, para un indicador que fuese: producción total de petróleo, la referencia sería '*petróleo*'.

Concepto: en el contexto de un indicador, describe la actividad, tarea, operación o cuenta realizada por el indicador abstrayéndolo de la referencia. Por ejemplo, para un indicador que fuese: producción total de petróleo, el concepto sería '*producción total*'.

4.3.2 Descripción general

4.3.2.1 Perspectiva del producto

La aplicación desarrollada pretende ofrecer las herramientas necesarias para la definición de indicadores energéticos, su clasificación y el posterior almacenamiento y consulta de los valores asociados.

Los usuarios podrán acceder a la aplicación desde cualquier sistema operativo que tenga conexión a Internet utilizando un navegador web.

Los sistemas de información podrán acceder a la aplicación desde cualquier sistema operativo que tenga conexión a Internet utilizando el protocolo http.

4.3.2.2 Funciones del producto

A continuación se enumeran las funcionalidades que se esperan de la aplicación:

- **Definir el sistema de clasificación:** Se podrá crear una clasificación arbitraria con un mínimo de dos niveles jerárquicos (secciones y subsecciones), que permitan la ordenación de los indicadores energéticos.
- **Definir indicadores energéticos:** Se podrá especificar indicadores energéticos, en los que se deberán definir los siguientes conceptos: tipo de indicador al que hace referencia el indicador y su concepto.
- **Definir localizaciones:** Se podrán generar localizaciones que servirán para asociar a un valor de un indicador energético un ámbito geográfico.
- **Definir árboles de localizaciones:** Se podrá organizar las localizaciones en árboles jerárquicos.
- **Definir unidades de medida:** Se podrán crear las unidades de medida que sean necesarias para la cuantificación de los valores de un indicador.
- **Almacenar valores de indicadores energéticos:** Se podrá almacenar un número ilimitado de valores de cada uno de los indicadores energéticos definidos en el sistema. Estos valores deberán hacer referencia a una

localización almacenada, a una unidad de medida almacenada. Todo valor tendrá una referencia temporal que podrá indicar no solo un instante preciso sino también un intervalo de tiempos.

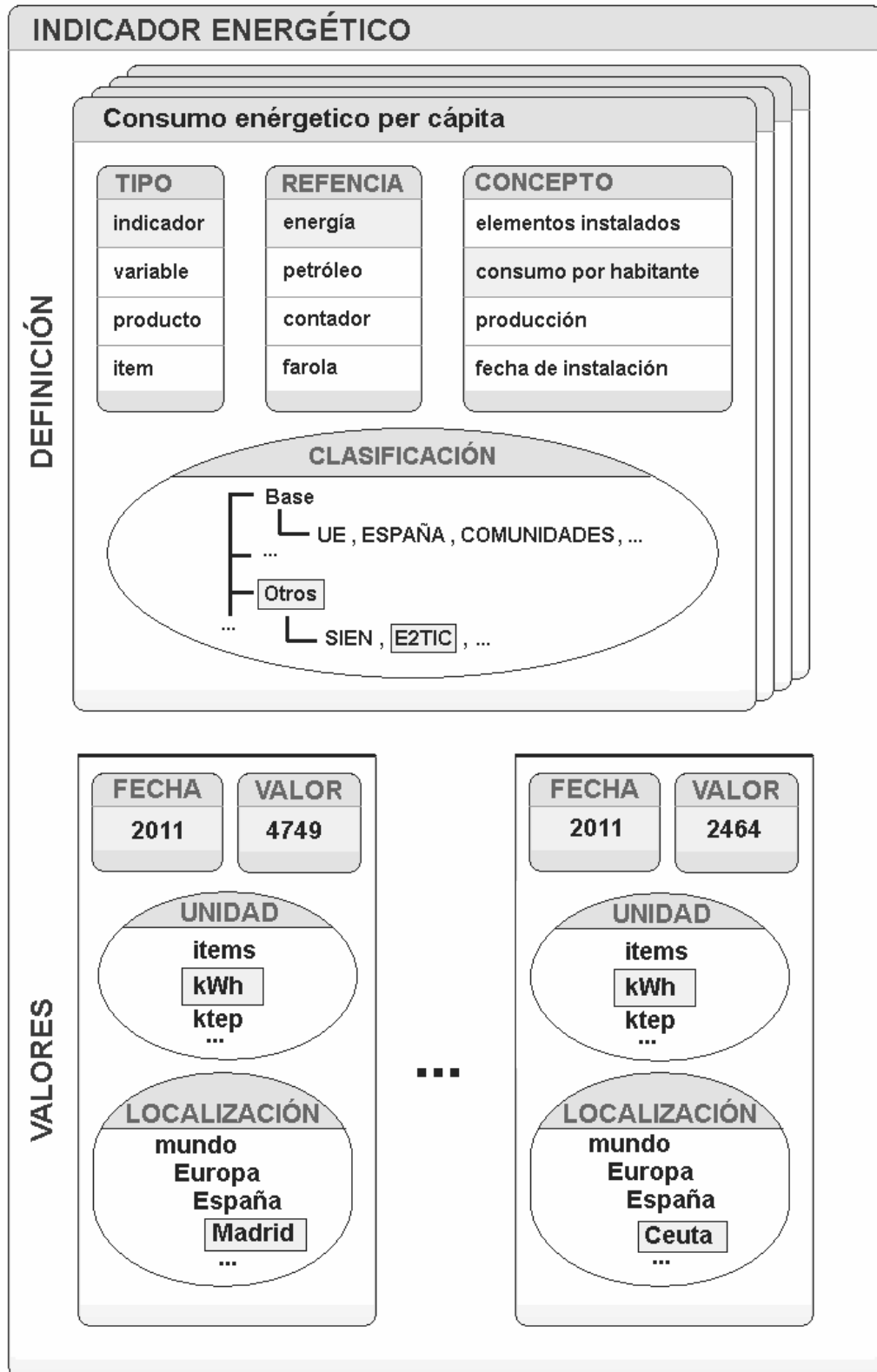


Figura 10 Requerimientos para la definición y valores de un indicador energético

4.3.2.3 Características de los usuarios

Se puede diferenciar entre dos tipos de usuarios:

- los usuarios humanos u operadores
- sistemas de información y/o dispositivos inteligentes.

4.3.2.3.1 Operadores

Este tipo de usuarios accederán al sistema a través de Internet usando un **navegador web**.

4.3.2.3.2 Sistemas de información/dispositivos inteligentes

Estos usuarios accederán al sistema a través de Internet usando el protocolo **http**. Cada una de las funciones implementadas para los operadores tendrá su equivalente en una **operación REST**.

4.3.2.4 Restricciones

Para el desarrollo de la aplicación se deberán tener en cuenta las siguientes restricciones de implementación:

- Lenguajes de programación: **PHP**
- Protocolos de comunicación: **http**
- Interfaces con otras aplicaciones: **REST**

4.3.2.5 Supuestos y dependencias

La aplicación desarrollada no requiere de ningún hardware o software específico, siendo capaz de funcionar en entornos **linux y windows**.

El servidor web donde sea alojado deberá soportar **PHP 5** o superior y tener conexión con un servidor de datos **MySQL 5** o superior.

La interfaz de usuario ha sido probada con un navegador **Firefox 23.0.1** y aunque se han seguido los estándares **W3C** es posible que la experiencia de usuario se vea

degrada con el uso de otros navegadores ya que algunos de ellos no siguen o no implementan correctamente los estándares **W3C**.

4.3.2.6 Requisitos Futuros

- Se implementará un **control de acceso** a los servicios.
- Se implementará un **sistema de roles** que permita restringir el acceso a los diferentes servicios.
- Se podrá **desactivar** tanto los indicadores como los **elementos** usados para su definición. Los elementos desactivados siguen existiendo pero no aparecen en los listados de selección.

4.3.3 Requisitos específicos

4.3.3.1 Interfaces externos

4.3.3.1.1 Interfaces de usuario

El código que genere la aplicación para ser visualizado a través de navegadores web debe seguir los estándares W3C.

4.3.3.1.2 Interfaces hardware

La aplicación podrá desplegarse sobre cualquier plataforma hardware.

4.3.3.1.3 Interfaces software

La aplicación funcionará sobre cualquier máquina con acceso a Internet.

4.3.3.1.4 Interfaces de comunicaciones

Las comunicación de la aplicación con otros sistemas se efectuará a través del protocolo HTTP usando conexiones TCP/IP.

4.3.3.2 Requerimientos funcionales

A continuación se presentan los diferentes requerimientos aplicables al sistema de gestión.

4.3.3.2.1 RQ-TL-01 - Crear Tipo de Localización.

Se podrán definir tipos de localizaciones, estos tipos serán usados para poder jerarquizar las localizaciones. Cada tipo de localización deberá tener asociada una descripción.

4.3.3.2.2 RQ-TL-02 - Modificar los tipos de localización.

Se podrá modificar la descripción de un tipo de localización.

4.3.3.2.3 RQ-TL-03 - Borrar Tipo de Localización.

El borrado solo debe ejecutarse si no existe ninguna localización que pertenezca a dicho tipo.

4.3.3.2.4 RQ-TL-04 - Listar los tipos de localizaciones.

Se generarán los listados, con la información y los filtros necesarios, para ser usados en la definición de localizaciones.

4.3.3.2.5 RQ-LC-05 - Crear Localizaciones.

Cada localización tendrá definido un tipo de localización y una descripción.

4.3.3.2.6 RQ-LC-06 - Modificar las localizaciones.

Se podrá modificar la descripción de una localización.

4.3.3.2.7 RQ-LC-07 - Borrar Localización.

El borrado solo debe ejecutarse si no existe ninguna localización descendente asociada.

RQ-LC-08 - Listar localizaciones

Se generarán los listados, con la información y los filtros necesarios, para ser usados en la generación de un árbol jerarquizado de localizaciones y en la construcción del valor de un indicador.

4.3.3.2.8 RQ-AL-09 - Crear árboles jerarquizados de localizaciones.

Cada localización como máximo aparecerá una vez en cada árbol. El tipo de localización se usará para la jerarquía del árbol. Una localización no podrá asociarse a otra del mismo tipo.

4.3.3.2.9 RQ-CC-10 - Crear clasificación de indicadores.

Se podrá definir una clasificación de indicadores con dos niveles. Al nivel superior le asociaremos el término **sección** y al segundo nivel **sub-sección**. Los indicadores podrán ser relacionados a una sección o a una sub-sección.

4.3.3.2.10 RQ-TI-11 - Crear Tipo de Indicador.

Se podrán definir tipos de indicadores, estos tipos serán usados para la definición de indicadores. Cada tipo de indicador deberá tener asociada una descripción.

4.3.3.2.11 RQ-TI-12 - Modificar los tipos de indicadores.

Se podrá modificar la descripción de un tipo de indicador.

4.3.3.2.12 RQ-TI-13 - Borrar Tipo de Indicador.

El borrado solo deberá ejecutarse si no existe ningún indicador que pertenezca a dicho tipo.

4.3.3.2.13 RQ-TI-14 - Listar los tipos de indicadores.

Se generarán los listados, con la información y los filtros necesarios, para ser usados en la definición de indicadores.

RQ-RI-15 - Crear Referencia

Se podrán definir referencias, éstas serán usadas para la definición de indicadores. Cada referencia deberá tener asociada una descripción.

4.3.3.2.14 RQ-RI-16 - Modificar las referencias.

Se podrá modificar la descripción de una referencia.

4.3.3.2.15 RQ-RI-17 - Borrar Referencia.

El borrado solo deberá ejecutarse si no existe ningún indicador que use dicha referencia.

4.3.3.2.16 RQ-RI-18 - Listar las referencias.

Se generarán los listados, con la información y los filtros necesarios, para ser usadas en la definición de indicadores.

4.3.3.2.17 RQ-CI-19 - Crear Concepto.

Cada concepto deberá tener asociado una descripción. Se podrán definir conceptos, estos serán usados para la definición de indicadores.

4.3.3.2.18 RQ-CI-20 - Modificar descripción de los conceptos.

Se podrá modificar la descripción de los conceptos.

4.3.3.2.19 RQ-CI-21 - Borrar Concepto.

El borrado solo deberá ejecutarse si no existe ningún indicador que use dicho concepto.

4.3.3.2.20 RQ-CI-22 - Listar los conceptos.

Se generarán los listados, con la información y los filtros necesarios, para ser usados en la definición de indicadores.

RQ-UM-23 - Crear Unidad de Medida.

Se podrán definir unidades de medida. Cada unidad de medida tendrá asociado un código, un nombre, una indicación de unidad base (en caso de no ser unidad base, se deberá especificar la referencia a la unidad base) y un factor de conversión.

4.3.3.2.21 RQ-UM-24 - Modificar Unidad de Medida.

Toda la información asociada a una unidad de medida es susceptible de ser modificada.

4.3.3.2.22 RQ-UM-25 - Borrar Unidad de Medida.

El borrado solo deberá ejecutarse si no existe ningún valor de indicador que use dicha unidad de medida.

4.3.3.2.23 RQ-UM-26 - Listar las unidades de medida.

Se podrán listar las unidades de medida, pudiendo ser ordenadas y filtradas, al menos, atendiendo a su código, nombre y/o unidad base.

4.3.3.2.24 RQ-IE-27 - Definir Indicador Energético.

Se podrán definir indicadores energéticos. Cada indicador se caracterizará por un tipo, una referencia y un concepto. El indicador dispondrá de una identidad numérica y tendrá asociado un código alfanumérico y una descripción. El indicador podrá ser asociado a una sección o sub-sección definida en una clasificación (ver requisito RQ-CC-10).

4.3.3.2.25 RQ-IE-28 - Modificar indicador.

Toda la información asociada a un indicador es susceptible de ser modificada con la excepción de su identidad numérica.

4.3.3.2.26 RQ-IE-29 - Borrar Indicador.

El borrado solo deberá ejecutarse si no existe ningún valor asociado al indicador.

4.3.3.2.27 RQ-IE-30 - Listar los indicadores.

Se podrán listar los indicadores pudiendo ser ordenados y filtrados al menos por su tipo, referencia, concepto y/o clasificación.

4.3.3.2.28 RQ-IE-31 - Almacenar valores de los Indicadores Energéticos.

El valor podrá ser decimal. Cada valor tendrá la referencia al indicador, a una unidad de medida y a una localización. El valor tendrá una referencia temporal con una precisión de segundos, adicionalmente podrá especificarse un intervalo de tiempo expresado también en segundos.

4.3.3.2.29 RQ-IE-32 - Borrar valores de los Indicadores Energéticos.

Se podrá borrar cualquier valor almacenado.

4.3.3.2.30 RQ-IE-33 - Listar valores de indicadores energéticos.

Se podrán listar los valores de los indicadores pudiendo ser ordenados y filtrados al menos por su indicador y/o su localización.

4.3.3.3 Rendimiento

La aplicación deberá comportarse de forma ágil de cara al usuario.

4.3.3.4 Atributos

4.3.3.4.1 Fiabilidad

Todas las operaciones que se realicen sobre el sistema devolverán un código de error o un aviso al usuario en caso de fallo.

4.3.3.4.2 Mantenibilidad

El despliegue y mantenimiento de la aplicación a nivel de sistemas se llevará a cabo por un el administrador. No deben existir operaciones de mantenimiento específicas de la aplicación.

4.3.3.4.3 Portabilidad

Los requisitos de la aplicación permiten su despliegue en cualquier plataforma y bajo cualquier sistema operativo.

4.4 Diseño

La función del Sistema de Gestión de Datos de Indicadores Energéticos (**SGDIE**) es proveer de las herramientas necesarias para la definición y posterior almacenamiento de cualquier indicador energético.

El sistema usa el **protocolo http** como vía de comunicación para la captura de información y los **navegadores web** (Firefox, Chrome, IE, etc.) como **interfaz de usuario**.

Para su posterior uso estos indicadores se clasifican siguiendo un criterio arbitrario definido por el usuario y sus valores son localizados para su agregación.

Este sistema se divide en cinco subsistemas, cuyas misiones se describen brevemente a continuación.

- **Subsistema Identidad Única (SIU)**: provee de identificadores únicos. Es un subsistema auxiliar que ofrece su servicio al resto de subsistemas.
- **Subsistema Clasificación (SCL)**: permite realizar una clasificaciones a dos niveles.
- **Subsistema Localizaciones (SLO)**: suministra localizaciones jerarquizadas.
- **Subsistema Árbol Jerárquico (SAJ)**: permite construir un árbol de localizaciones partiendo de una fuente de éstas. Su principal objetivo es tener diferentes vistas usando las mismas localizaciones.
- **Subsistema Gestión de Indicadores (SGI)**: introduce dinámicamente indicadores en el sistema y deposita los valores de los indicadores suministrados por terceros para su posterior consulta.

En la siguiente figura se muestra un esquema del sistema. Cada subsistema hace uso de una base de datos independiente.

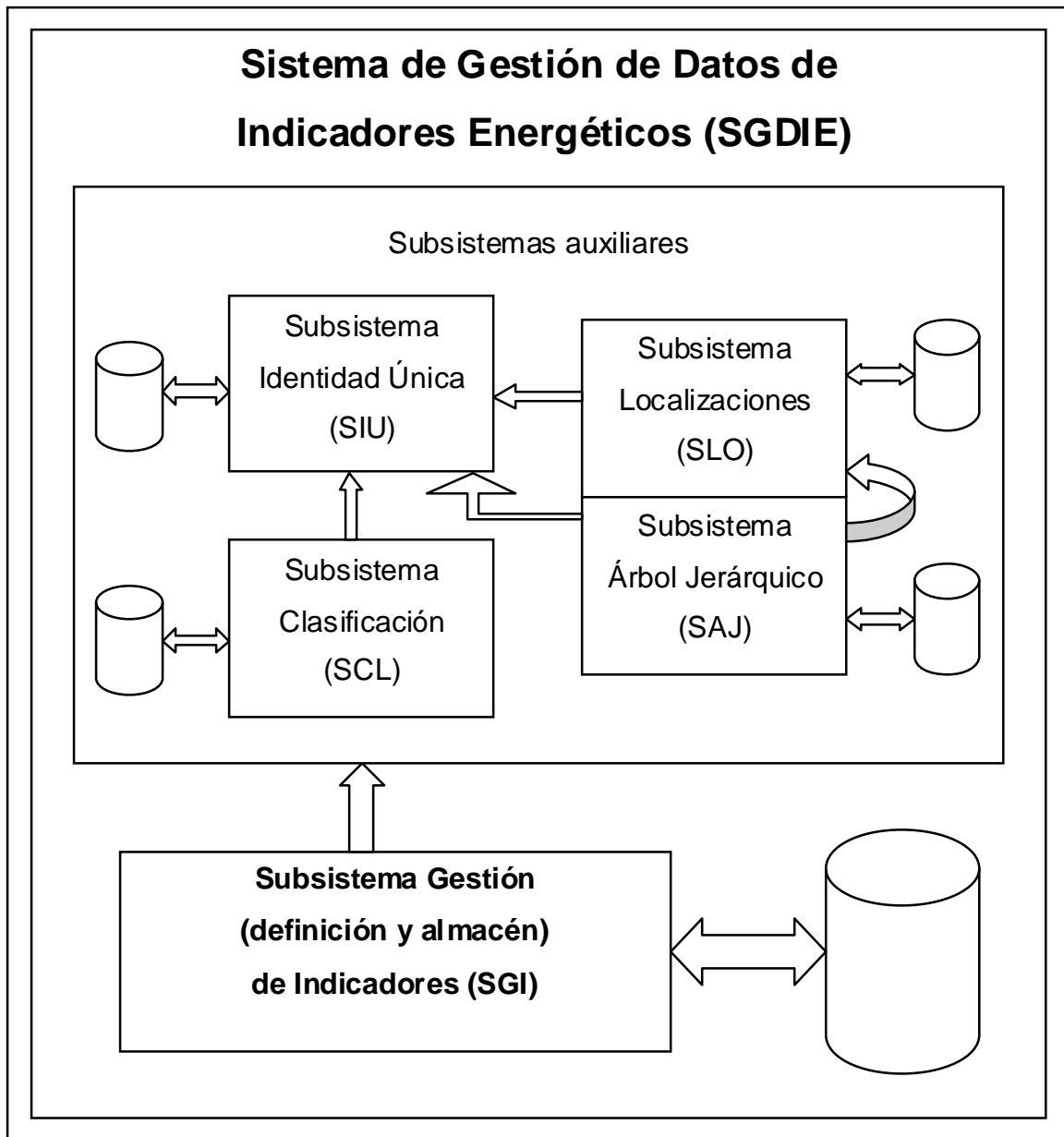


Figura 11 Esquema del Sistema de Gestión de Datos de Indicadores Energéticos

4.4.1 Modelo de datos

4.4.1.1 Subsistema Identidad Única

Para implementar el servicio identidad única (svcid), hemos diseñado una tabla en la cual se almacena la última identidad dada (campo XID) para cada tipo de identidad (campo XID_TYPE)

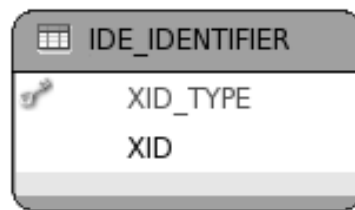


Figura 12 Esquema de datos del subsistema 'identidad única'

En la implementación hemos prefijado la tabla con "xid_".

4.4.1.2 Subsistema Localizaciones

Los diferentes tipos de localizaciones se almacenan en la tabla LOT_LOCATION_TYPE, estos tipos califican de forma jerárquica las diferentes localizaciones que son almacenadas en la tabla LOC_LOCATION.

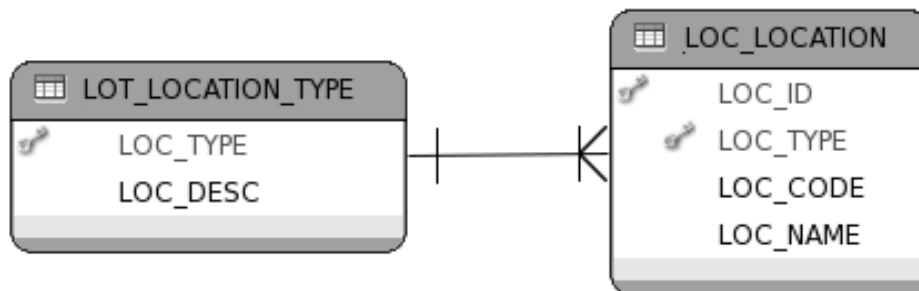


Figura 13 Esquema de datos del subsistema 'localizaciones'

En la implementación se ha prefijado las tablas con "spot_".

4.4.1.3 Subsistema Árbol Jerárquico

La identidad de los árboles se almacena en la tabla HTT_HTREE_TREE, la jerarquía de niveles en la tabla HTL_HTREE_LEVEL y los nodos en HTN_HTREE_NODE.

La información acerca de la estructura se almacena en HTF_HTREE_FOREST. Esta tabla puede contener datos de múltiples árboles cada uno de ellos identificado por el campo HTR_TREE_ID. En esta estructura cada nodo tiene un nivel jerárquico indicado en el campo HTR_LEVEL y al hacer la inserción de datos se comprueba que este nivel siempre sea mayor que el del nodo padre referenciado. La tabla contiene a la vez una estructura de lista adyacente, una estructura de elementos jerarquizados y otra lineal.

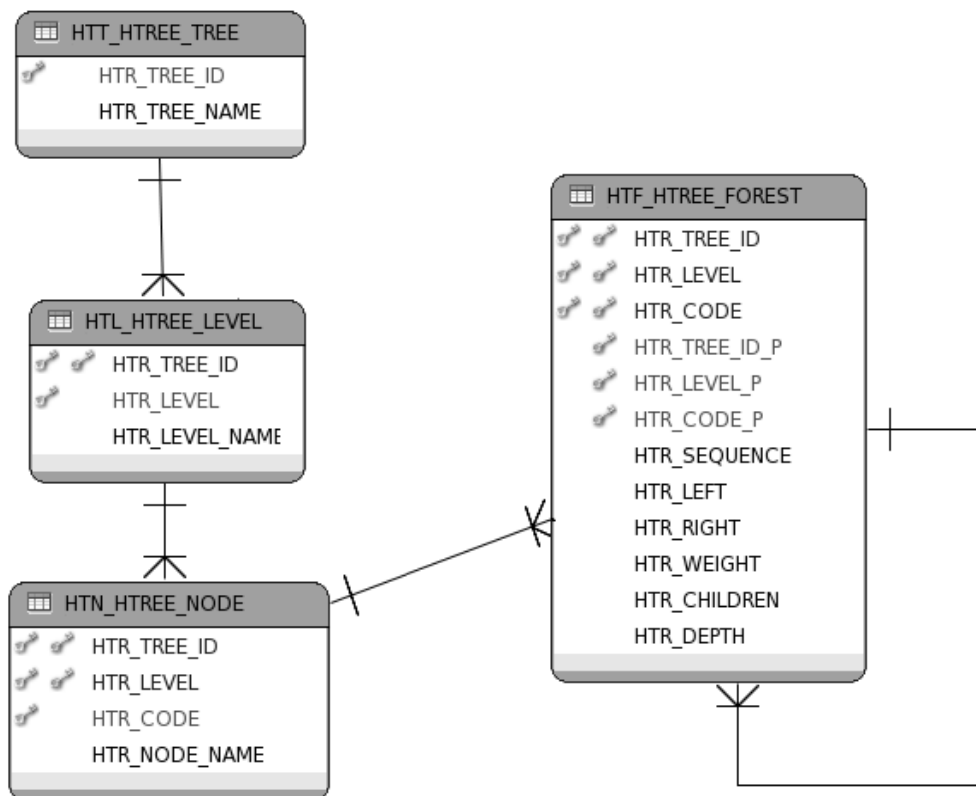


Figura 14 Esquema de datos del subsistema 'árbol jerárquico'

En la implementación se ha prefijado las tablas con "forest_".

4.4.1.4 Subsistema Clasificación

Este esquema es muy parecido al usado para árboles de localizaciones. La clasificación tiene una identidad que es almacenada en la tabla RTK_RATING_KIND, la jerarquía de niveles (clasificación, sección y subsección) en la tabla RTL_RATING_LEVEL y los nodos en RTN_RATING_NODE.

Cada clasificación de nodos jerarquizada se refleja en RTV_RATING_VIEW. Estas tablas pueden contener datos de múltiples clasificaciones identificadas a través del campo HTG_RATING_ID. Al igual que para los árboles, cada nodo tiene un nivel jerárquico indicado en el campo HTR_LEVEL y al hacer la inserción de datos se comprueba que este nivel siempre sea mayor que el del nodo padre referenciado.

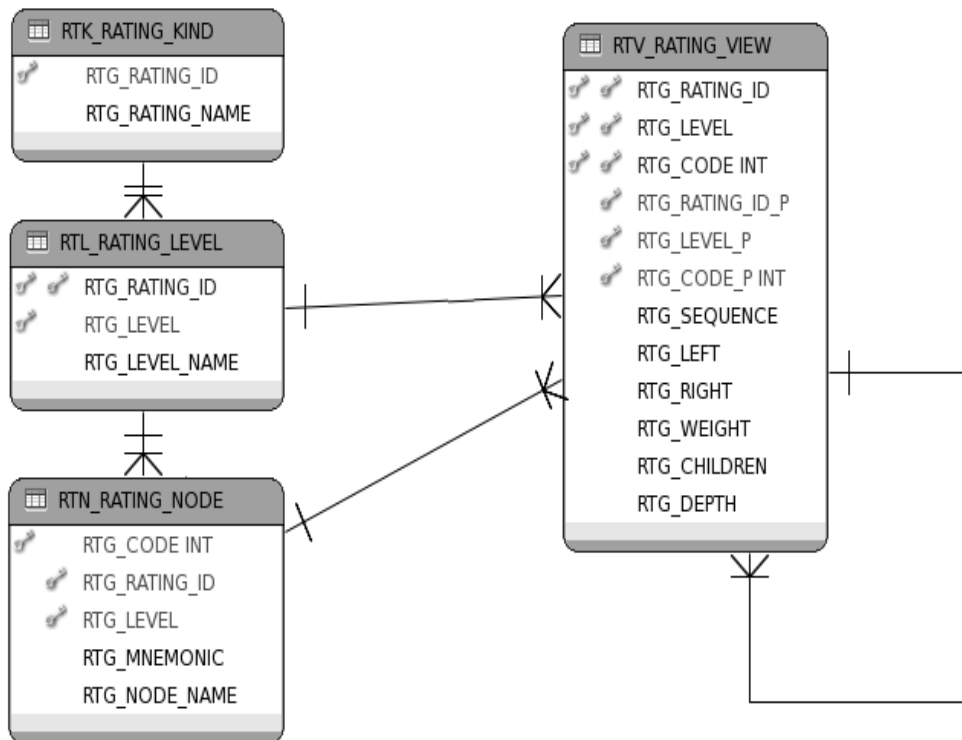


Figura 15 Esquema de datos del subsistema 'clasificación'

En la implementación se ha prefijado las tablas con "chapter_".

4.4.1.5 Subsistema Gestión de Indicadores

Esta es la estructura principal del sistema y se explica con detalle en los siguientes apartados.

El campo RTG_CODE de la tabla CND_COUNTER_DEFINITION hace referencia al mismo campo de la tabla RTN_RATING_NODE del subsistema de clasificación. El campo LOC_ID de la tabla CNV_COUNTER_VALUE hace referencia al campo HTR_CODE de la tabla HTN_HTRE_NODE del servicio de árboles de localizaciones.

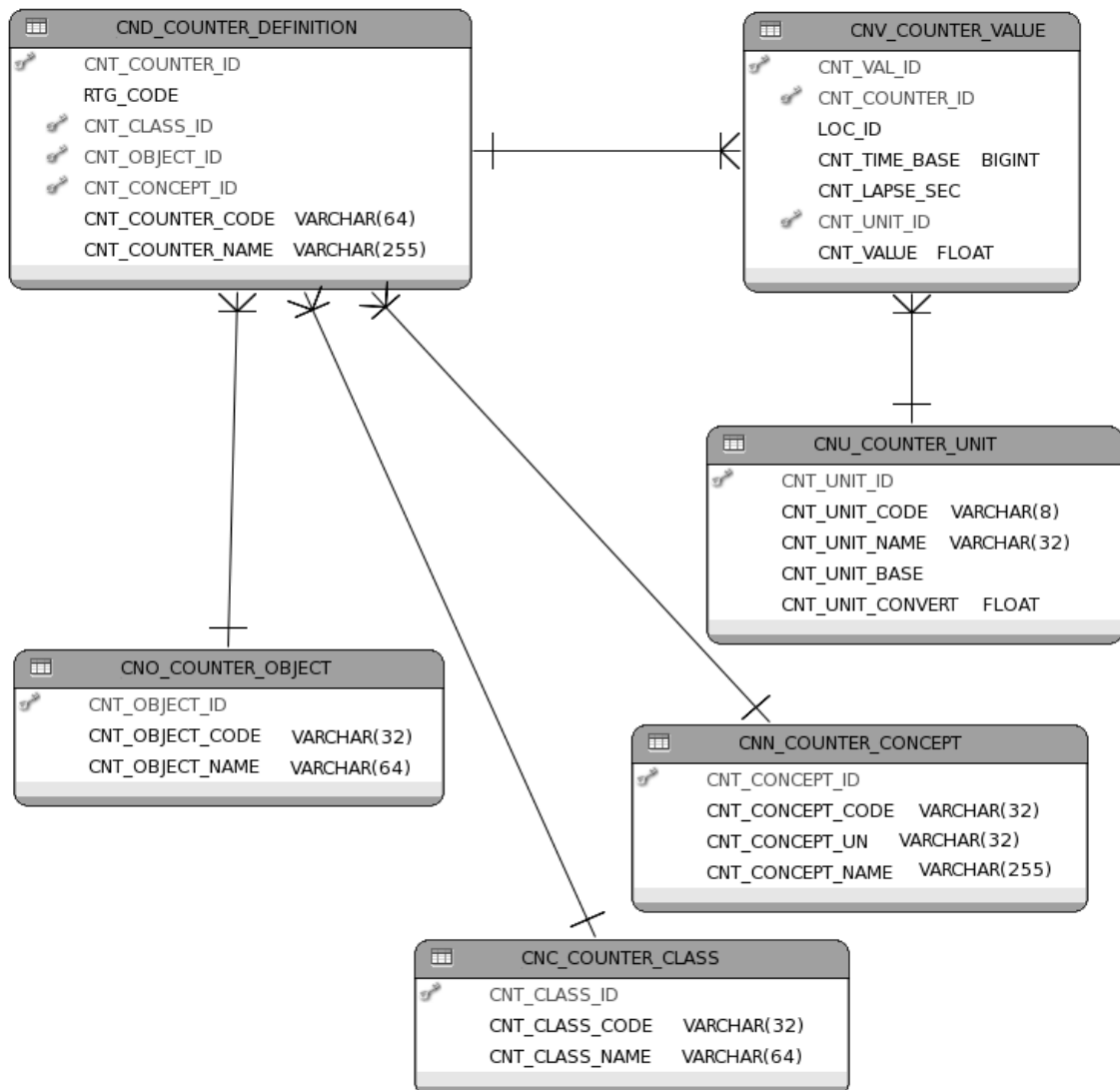


Figura 16 Esquema de datos del subsistema 'gestión de indicadores'

En la implementación hemos prefijado las tablas con "cdef_".

4.4.1.6 Atributos de la definición del Indicador

A continuación se detallan de los atributos que definen a un indicador:

Identidad (CNT_COUNTER_ID)

Identificación del indicador. Es un código arbitrario y único en el sistema.

Clasificación (RTG_CODE)

Este atributo permite la clasificación de los indicadores para su tratamiento por personas (ordenación, selección, etc.). Esta clasificación contempla dos niveles: secciones y subsecciones.

Ejemplos de secciones: "*Energy Statistics Database*", "*Global Indicator Database*"

Ejemplos de subsecciones: dentro de la sección "*Global Indicator Database*": "*Greenhouse Gas*", "*Manufacturing*", "*Population*", etc

Tipo (CNT_CLASS_ID)

Describe un tipo de indicador. Ésta sirve para procesar de forma diferenciada las diferentes clases de indicadores.

Ejemplos de clase: "*Variable*", "*Commodity*", "*Indicator*", "*Item*"

La clase permite distinguir, por ejemplo, cuándo los indicadores pueden ser agregados por zonas (consumos eléctricos, costes) o cuándo no (estados).

Referencia (CNT_OBJECT_ID)

Este atributo define sobre qué vamos a contar.

Ejemplos: "*Aparatos de aire acondicionado*", "*Farolas*"

Existirán indicadores que nos permitan saber el número de objetos existentes, otros indicadores definirán consumos eléctricos de estos objetos, otros permitirán conocer el coste o la fecha de instalación.

Concepto (CNT_CONCEPT_ID)

Define lo que se pretende contar.

Ejemplo: para un indicador de clase "*Commodity*", cuya Referencia sea "Petróleo", existirán diferentes conceptos: "importaciones", "exportaciones", "consumo por la industria", "consumo por la agricultura", etc.

Una vez definido un indicador podemos obtener muestras del indicador.

Código (CNT_COUNTER_CODE)

Es un mnemónico del indicador.

Nombre (CNT_COUNTER_NAME)

Descripción del indicador.

4.4.1.7 Atributos del valor de un indicador

Cada uno de los valores asociados a un indicador posee los siguientes atributos:

Identidad (CNT_VAL_ID)

Identificación del valor. Es un código arbitrario y único.

Definición (CNT_COUNTER_ID)

Es un código que referencia a la definición del indicador.

Localización (LOC_ID)

Identifica al indicador con un punto espacial lo que permite agregar dichos valores por zonas geográficas.

Ejemplo: si están almacenados los valores de consumo energético de Madrid, Albacete, Cuenca, etc. se puede derivar el consumo de España.

Tiempo (CNT_TIME_BASE)

Referencia temporal a la que el indicador hace referencia. Se usa *unix timestamp* con valores negativos para fechas anteriores a 1970.

Intervalo (CNT_LAPSE_SEC)

El indicador puede hacer referencia a un intervalo de tiempo relativo a la base definida como "Tiempo". En este caso se especificará un intervalo en segundos.

Unidad de Medida (CNT_UNIT_ID)

Cada valor debe especificar su unidad de medida. Estas unidades están tabuladas y referencias a una base.

Valor (CNT_VALUE)

Estrictamente un número (entero o decimal).

4.4.2 Interfaces

Para el acceso a los servicios ofrecidos por los subsistemas se hace uso de tres interfaces para interactuar con la base de datos: XMLRPC, REST y PHP (API).

4.4.2.1 XMLRPC

Todos los parámetros de las peticiones se agrupan bajo una estructura en la que el parámetro '**op**' indica la operación a realizar y siempre está presente, adicionalmente se pueden incluir uno o más parámetros.

```
<?xml version="1.0"?>
<methodCall>
<methodName>xid.identifier</methodName>
<params>
  <param><value><struct>
    <member><name>op</name><value><string>get</string></value></member>
    <member><name>field_type</name><value><string>1</string></value></member>
  </struct></value>
</param>
</params>
</methodCall>
```

Figura 17 Ejemplo de petición XMLRPC

Todas las respuestas son estructuras con dos campos:

- **'error'**: (1:0) indicando si hubo error o éxito en la operación.
- **'data'**: el contenido del campo dependerá de la operación.

```
<?xml version="1.0"?>
<methodResponse>
<params>
  <param>
    <value><struct>
      <member><name>error</name><value><int>0</int></value></member>
      <member><name>data</name>
        <value><struct>
          <member><name>xid</name><value><string>1</string></value></member>
        </struct></value>
      </member>
    </struct></value>
  </param>
</params>
</methodResponse>
```

Figura 18 Ejemplo de respuesta XMLRPC

4.4.2.2 REST

Convierte las URL en peticiones **XMLRPC**. El formato de las URL de estas peticiones es:

[http://\[host\]/\[método\]/\[operación\]/\[formato\]/?\[parámetros\]](http://[host]/[método]/[operación]/[formato]/?[parámetros])

Los formatos disponibles son: *js (json)*, *text*, *php* y *xml*. Todas las operaciones XMLRPC pueden ser ejecutadas con este interfaz.

4.4.2.3PHP

Se convierten las llamadas a métodos en peticiones XMLRPC. Sólo las operaciones que necesitan un subsistema de otro subsistema se definen en este interfaz o API.

4.5 Casos de uso

Para validar el sistema se han simulado datos para los casos de uso descritos a continuación.

4.5.1 Indicadores energéticos definidos por E2TIC

El E2TIC es el Clúster de Eficiencia energética y TIC de la red de cátedras de Telefónica.

En el documento “*Eficiencia Energética y TIC*” se presentan un total de 6 indicadores de eficiencia energética de relevancia en el estado español. Estos indicadores pretenden cubrir todos los sectores dónde las TIC se han perfilado como solución para mejorar la eficiencia energética [20].

Para el seguimiento de los indicadores presentados en el documento anterior, se almacenan datos anuales por comunidades autónomas y se consultan dichos datos a nivel nacional:

- Consumo energético per cápita en kWh².
- Proporción de consumo de energías renovables expresado en porcentaje, como resultado de la expresión: total consumo de renovables (ktep³) / consumo total de energía (ktep).
- Unidades vendidas de vehículos eléctricos.
- Número existente de ciudades inteligentes.

² kWh: kilovatio-hora

³ Ktep: mil toneladas equivalentes de petróleo.

- Número existente de edificios inteligentes.
- Número existente de centros con logística inteligente.

Para realizar el seguimiento se presenta una tabla por indicador indicado el valor del indicador a nivel nacional y el año al que pertenece dicho valor.

4.5.2 Seguimiento del consumo del sistema de luminarias en una administración local.

Una administración local de una Comunidad autónoma tiene instalados tres sistemas de alumbrado diferentes en tres calles diferentes. Cada luminaria posee un contador inteligente que reporta su consumo a un centro de control, éste a su vez vuelca mensualmente el consumo eléctrico de cada dispositivo en el banco de datos.

- a) Para realizar el seguimiento del consumo eléctrico en cada una de las calles se visualiza una tabla en la que se refleja el consumo eléctrico mes a mes de la calle para el año especificado.
- b) Para determinar el sistema de alumbrado más eficiente se hace un seguimiento del consumo medio por luminaria de cada uno de los sistemas. Para ello se genera un informe en el que para el año indicado se visualiza el consumo medio de las luminarias de cada uno de los sistemas.

4.6 Validación

Para validar el sistema se han simulado datos y se ha generado las aplicaciones necesarias para reflejar los casos de uso descritos en el punto anterior.

5 Planos

No se ha realizado ningún plano en el desarrollo del proyecto.

6 Presupuesto

Para la realización del siguiente presupuesto sólo se ha tenido en cuenta la mano de obra, si se necesitase desplegar el prototipo desarrollado habría que incrementar los costes teniendo en cuenta el alojamiento de la aplicación web.

Detalle de la mano de obra:

Especificación de de requisitos:

3 personas/día de analista = 24 * 60€ = 1440€

Diseño:

5 personas/día de analista/programador = 40 * 50€ = 2000€

Programación:

10 personas/día de un programador = 80 * 40€ = 3200€

Pruebas y validación:

2 personas/día de analista = 16 * 60€ = 990€

Total: 7.630 €

7 Manual de Usuario

7.1 Guía de instalación

Se describe a continuación los pasos necesarios para instalar la aplicación en un entorno Windows. Estas instrucciones han sido probadas sobre un ordenador personal con un sistema operativo Windows7 64 bits.

Las instrucciones **se indican a modo de orientación**, el administrador del sistema debe adaptarlas a su entorno al realizar la instalación.

- a) Crear una carpeta en la unidad c:

```
mkdir c:\pfc_sgdie
```

- b) Copiar el contenido completo del cd-rom en la carpeta creada.

- c) Crear una unidad virtual r: sobre el directorio c:\pfc_sgdie\extras

```
SUBST R: c:\pfc_sgdie\ extras
```

- d) Iniciar el servidor de datos, mysql, ejecutando el archivo de lotes:

```
r:\mysql_start.bat
```

- e) Crear las siguientes bases de datos con codificación de caracteres utf-8:

```
toolkit_cdef, toolkit_chp, toolkit_db, toolkit_loc, toolkit_pfc,  
toolkit_tree, toolkit_xid
```

- f) Importar los datos ejecutando las sentencias sql de los siguientes ficheros, que se encuentran en la carpeta r:\data:

```
toolkit_db_full.sql, toolkit_xid_full.sql, toolkit_loc_full.sql,  
toolkit_tree_full.sql, toolkit_chp_full.sql, toolkit_cdef_full.sql
```

g) Crear un usuario 'toolkit', con password 'pfc' en la base de datos con acceso a través del host 'localhost' y que tenga permisos universales sobre las bases de datos creadas.

h) Iniciar el servidor web, apache, ejecutando el archivo de lotes:

```
r:\apache_start.bat
```

i) Comprobar desde un navegador web que se puede acceder a la aplicación desde la url:

```
http:\localhost
```

7.2 Guía de administración

Si se han cargado los datos referidos en la guía de instalación, el sistema contará con un usuario administrador con el alias **toolkit** y contraseña: **pfc**.

La aplicación no necesita ningún tipo de administración específica.

Al estar desarrollada sobre un entorno drupal, todas las guías de administración de este entorno son de aplicación.

Se recomienda consultar los contenidos de la guía de 'Instalación y configuración' de Drupal Hispano a través de la siguiente url:

```
http://drupal.org.es/manuales/manual_del_administrador
```

7.3 Guía de usuario

La aplicación en su página inicial presenta un aspecto como el que muestra la siguiente figura:

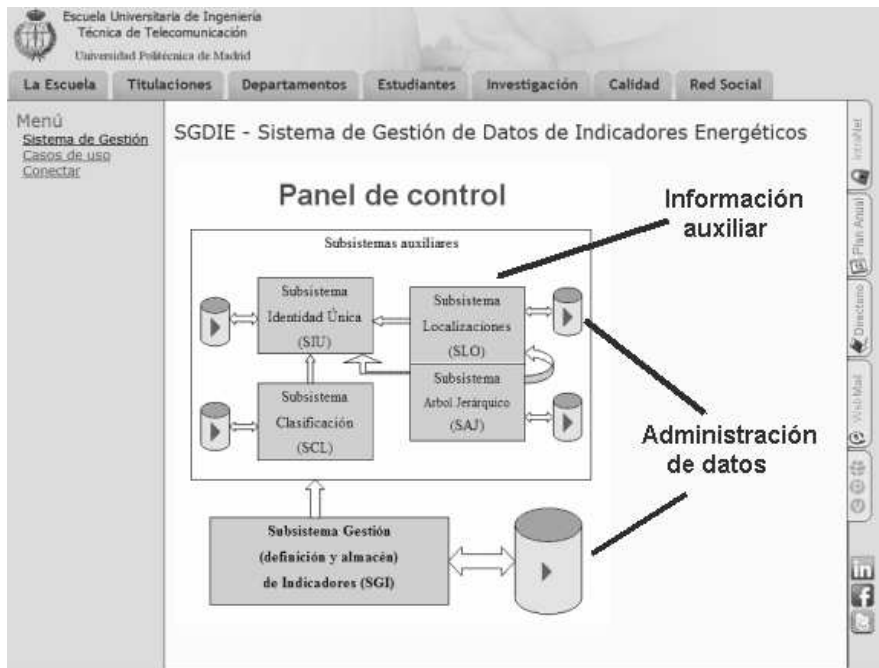


Figura 19 Página inicial de la aplicación.

A través de esta página el usuario puede:

- Conectarse al sistema a través del enlace titulado 'conectar'.
- Acceder a **herramientas de ayuda al desarrollo** de los diferentes subsistemas pulsando sobre las respectivas cajas verdes, señaladas en la figura anterior como 'Información auxiliar'.
- Acceder a las **herramientas de manipulación de datos**, que son objeto de la aplicación, pulsando sobre las figuras que representan bases de datos y que han sido señaladas en la figura anterior con la etiqueta 'Administración de datos'.

7.3.1 Herramientas de ayuda al desarrollo

En esta área página encontraremos información del subsistema de utilidad para el desarrollo, y unos enlaces que nos permitirán acceder a una herramienta que nos permitirá probar los procedimientos XMLRPC existentes. En la siguiente figura se muestra el aspecto que tiene,

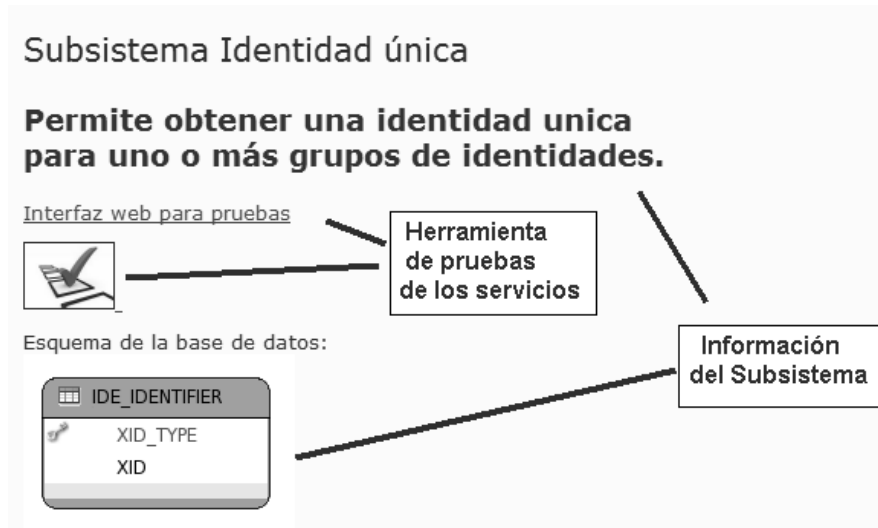


Figura 20 Página de herramientas de ayuda al desarrollo.

7.3.1.1 Herramienta de pruebas de los servicios.

En esta herramienta podremos seleccionar cada uno de los métodos y operaciones implementados. Se puede elegir diferentes formatos de salida a través del enlace con el título '*formats*'.

Una vez introducidos los parámetros deseados en los campos de formulario que estén configurados, lanzaremos la petición pulsando el botón de ejecución. El resultado será visualizado en el área superior destinado al efecto con una indicación de si la petición ha sido procesada correctamente o si se ha detectado algún error.

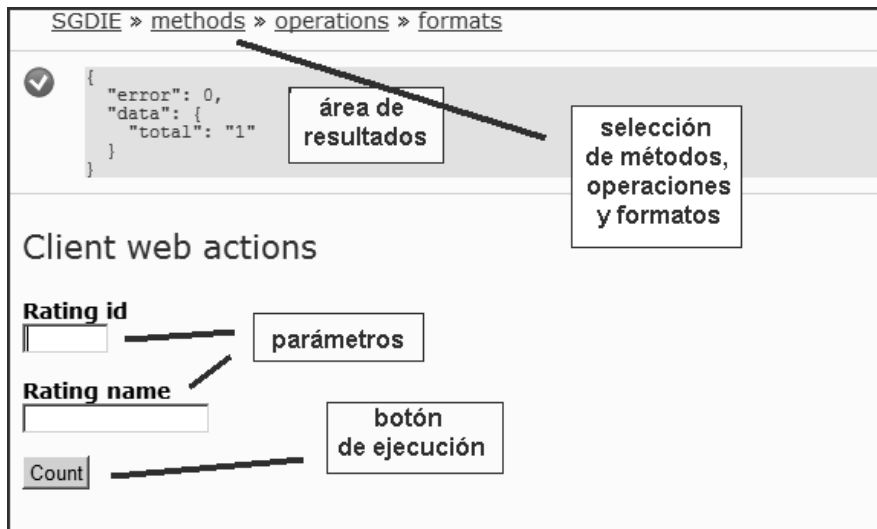


Figura 21 Herramienta de pruebas de los servicios.

7.3.2 Herramientas de manipulación de datos

Toda la funcionalidad objeto de la aplicación se accede a través de dos elementos:

- Tablas CRUD
- Árboles de selección y clasificación

7.3.2.1 Tablas CRUD

Estas tablas permiten el filtrado de datos, así como la creación, borrado y modificación de los mismos.

Si un campo se puede modificar, basta con hacer clic sobre el dato y éste se convertirá en un elemento activo que nos permitirá modificar el valor.

Nodos del árbol (1) United Nations

Mostrar 10 registros		Add	Delete	Buscar:
Nivel #	Code #	Niv		Nombre del Nodo
1	900	Global group	5	World Land
20	935	Mayor area		Asia
20	903	Mayor area		Africa
30	923	Region		Eastern Europe
30	924	Region		Northern Europe
30	925	Region		Southern Europe

Nivel #	Code #	Niv	Nombre del Nodo
---------	--------	-----	-----------------

Registros 1 a 10 de 257 Primero Anterior 1 2 3 4 5 Siguiete Último

Figura 22 Tabla CRUD

Podemos distinguir los siguiente elementos:

- 1) Selección de número de filas a mostrar
- 2) Botón de inserción de filas. Se presentará un formulario que nos permitirá insertar nuevas filas.
- 3) Botón de borrado. Elimina la fila seleccionada.
- 4) Búsqueda global. La cadena introducida buscará en los campos disponibles.
- 5) Control de ordenación por campo.
- 6) Búsqueda individual. La cadena introducida buscará unicamente en la columna correspondiente.
- 7) Control de paginación.

7.3.2.2 Árboles de selección y clasificación

Con esta herramienta, podemos seleccionar y clasificar diferentes elementos. El mecanismo para mover un elemento consiste en pulsar con el ratón sobre el mismo y soltarlo en la posición deseada, Los nuevos elementos están disponibles en la

carpeta denominada 'Nodos huérfanos'. Para eliminar un elemento de la estructura debemos seleccionarlo y arrastrarlo hasta la carpeta mencionada anteriormente.

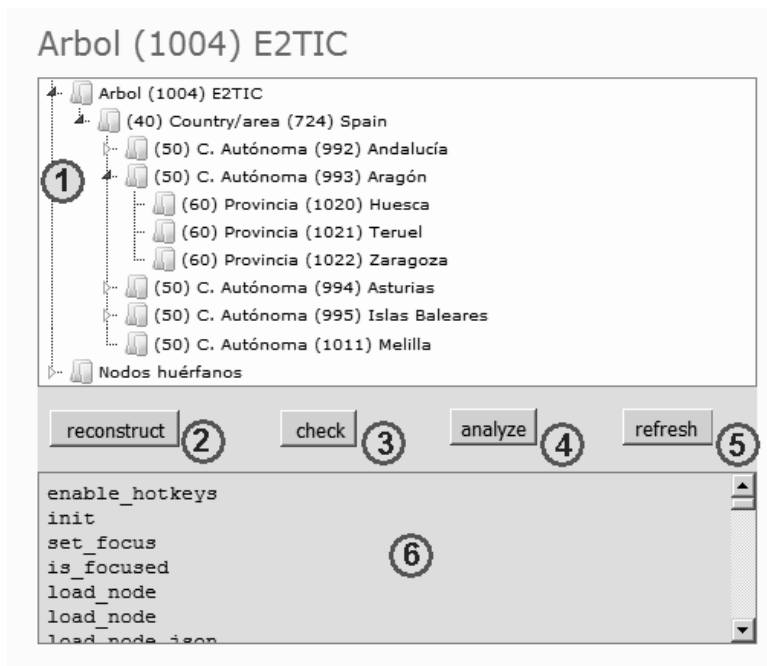


Figura 23 Árboles de selección y clasificación

Podemos distinguir los siguiente elementos:

- 1) Ventana de selección y clasificación.
- 2) Botón de reconstrucción. Reconstruye la información de la estructura de conjuntos anidados partiendo de la lista adyacente.
- 3) Botón de chequeo, Comprueba que la información suministrada por la estructura de conjuntos anidados y la de lista adyacente es la misma..
- 4) Botón de análisis. Realiza diversos chequeos de integridad en la estructura de conjuntos anidados
- 5) Botón de refresco. Recarga la información mostrada en el árbol.
- 6) Ventana con información de depuración.

8 Conclusiones

El banco de datos universal de indicadores energéticos es un gran paso para el aumento de la eficiencia energética aunque su mayor contribución es servir de repositorio de datos a aplicaciones expertas capaces de:

- **Monitorizar** los indicadores energéticos almacenados.
- **Mantener** dispositivos y sistemas mediante el análisis y la detección de patrones anómalos.
- **Simular** el efecto que produciría sobre el consumo energético, la actuación sobre alguno de los indicadores almacenados.
- **Predecir** el consumo energético atendiendo a diferentes factores (como por ejemplo el crecimiento estimado de población).

Para aumentar la eficiencia energética es imprescindible el desarrollo de **herramientas** que evalúen de forma **sistematizada** el consumo energético, **examinando** los indicadores generados por el sistema bajo observación y **comparando** los mismos con otros.

Estas herramientas permiten percatarse de la **situación real** e identificar de forma correcta los problemas de exceso de consumo eléctrico a la vez que ayudan a tomar decisiones más acertadas tanto para la **economía** como para el **medio ambiente**.

Ventajas del almacenamiento y monitorización de los indicadores energéticos:

- Determinar la **eficiencia en el consumo** de la energía eléctrica
- **Disminución** del consumo energético y por ende de la **facturación**.
- **Identificar los dispositivos** que contribuyen en mayor medida al consumo de energía.
- **Comparar la eficiencia** energética de los sistemas **con los niveles exigidos** y/o recomendados.
- **Estudiar cambios** que permitan realizar un uso más eficiente de la energía.

Un aspecto importante no abordado en la aplicación es la gestión de accesos por parte de los usuarios. Parece claro que deberían existir diferentes roles a la hora de operar sobre el sistema, con al menos un rol de administrador diferenciado. Las funciones relativas a la definición de los indicadores deberían de estar limitada a usuarios administradores. En la misma línea el borrado de datos debería de limitarse al usuario que aportó el valor o a los usuarios administradores.

9 Anexo I: Indicadores de Eficiencia Energética

Los siguientes indicadores han sido extraídos del documento “**Eficiencia energética y TIC**” creado por el Clúster de Eficiencia energética y TIC (E2TIC) de la red de cátedras Telefónica.

El Clúster E2TIC es una iniciativa conjunta de cinco cátedras universitarias:

Universidad Politécnica de Catalunya

Universidad de Oviedo

Universidad Politécnica de Valencia

Universidad de Extremadura

Universidad de Sevilla.

El objetivo del Clúster E2TIC es supervisar y promover el estado de avance de la diferentes iniciativas TIC en el territorio español dirigidas a cumplir con los objetivos de eficiencia energética marcados por la UE "20-20-20".

En el documento se presentan una serie de indicadores de eficiencia energética de relevancia en España. Estos indicadores pretenden cubrir sectores dónde las TIC se han perfilado como solución para mejorar la eficiencia energética.

A continuación se recogen algunos indicadores reflejados en dicho documento.

9.1 Consumo energético per cápita

El consumo total de energía per cápita mide la cantidad de la energía primaria consumida, en promedio, por cada persona que vive en un país o región en particular para el año indicado. Incluye todas las fuentes primarias de energía, incluidos el carbón, productos derivados del carbón, petróleo, productos derivados del petróleo, gas natural, energía nuclear, hidroeléctrica, etc.

España presenta un consumo anual de 34MWh, ligeramente por debajo del consumo europeo (38MWh) o de la OECD (51MWh), dos veces más que la media mundial (19MWh) y claramente por debajo del consumo de EE.UU. (83MWh).

Hay que destacar la dependencia de España del petróleo.

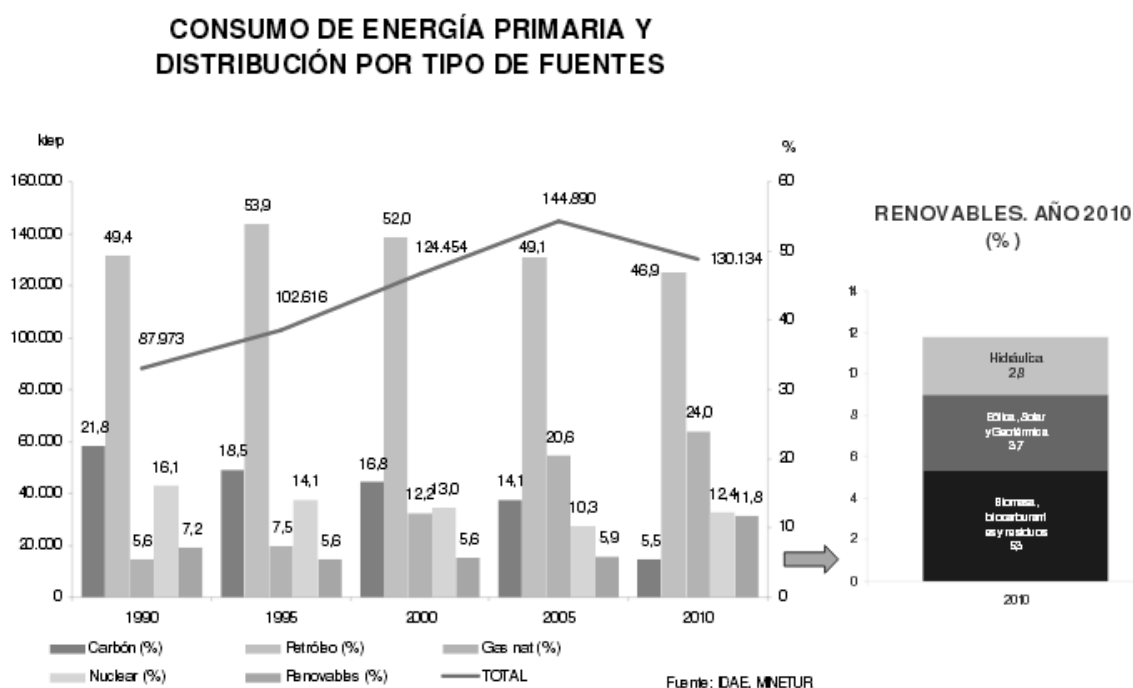


Figura 24 Uso de las diferentes fuentes de energía en España [21]

9.2 Proporción de Consumo de Energías Renovables

La cuota global de energías renovables en el consumo final de energía, es la relación entre el consumo interior bruto de energía procedente de fuentes renovables (TOE) y consumo interior bruto total de la energía (TOE), calculado para un año y expresada en porcentaje. Tanto la energía renovable como el consumo total de energía se miden en miles de toneladas equivalentes de petróleo (ktep).

Las energías renovables se definen como de fuentes de energía renovables no fósiles:

- energía eólica
- solar
- geotérmica
- del oleaje
- mareomotriz
- hidráulica
- biomasa
- gases provenientes de residuos
- biogás

En los últimos años, España se ha posicionado como uno de los líderes en energías renovables, es el país con más crecimiento en la UE y la octava a nivel mundial.

Cuando se habla de consumo de energías renovables conviene diferenciar entre el consumo primario y el consumo de energías renovables para la producción de electricidad.

La Comisión Europea decretó una directiva comunitaria (2009/28/CE) que contempla los objetivos obligatorios de consumo de energías renovables para la UE y para cada

uno de los Estados miembros en el año 2020, y la elaboración por parte de éstos, de planes de acción nacionales para alcanzar dichos objetivos.

Consumo Final de Energía (ktep)	2012	2016	2020
Consumo de energía bruta final	93.321	95.826	98.677
Energías renovables para generación eléctrica	8.477	10.682	13.495
Energías renovables para calefacción /refrigeración	3.955	4.740	5.618
Energías renovables en transporte	2.073	2.786	3.500
Total en renovables en ktep	14.504	18.208	22.613
Total en renovables según Directiva	14.505	17.983	22.382
% Energías renovables / Energía final	15.5%	18.8%	22.7%

Tabla 1 Consumo final de energía en España según el Plan de Renovables 2011-2020

España presentó un anticipo del Plan de Renovables 2011-2020, en una primera estimación, la aportación de las energías renovables al consumo final bruto de energía sería del 22,7% en 2020, siendo el objetivo para España del 20% en 2020. La previsión de la contribución de las energías renovables a la generación bruta de electricidad en España es del 42,3% en 2020, con lo que se superará el objetivo fijado por la UE en este ámbito (40%).

9.3 Vehículos eléctricos

La definición de coche eléctrico abarca los coches completamente eléctricos con batería y automóviles híbridos “*plug-in*”. El concepto de coches eléctricos está fuertemente ligado a la campaña mundial para la reducción de los niveles de CO2 mediante la generación de energía sin dependencia de los combustibles fósiles.

La mayoría de los gobiernos están fomentando el mercado de automóviles eléctricos, las industrias relacionadas, la implantación de una infraestructura de carga de la batería, así como incentivos a los consumidores.

Ventas coches eléctricos en España	2011	2012
Coche híbrido	10.350	10.030
Coche eléctrico	377	484
% Ventas coche híbrido + eléctrico / total	1.26%	1.44 %

Tabla 2 Ventas coches eléctricos en España [22]

Se pronostica que el mercado europeo de coches eléctricos crecerá un 34% entre 2011 y 2020.

Se estima que a finales de la década, las matriculaciones de vehículos eléctricos en España representen un 14% del total de matriculaciones.

9.4 Ciudades Inteligentes

El desarrollo de la eficiencia energética en el marco de las “*smart cities*” se centra en los aspectos siguientes:

- a) eficiencia en la red de distribución,
- b) reducción de la demanda en los picos,
- c) mejora de la relación pico-media en la curva de demanda diaria,
- d) reducción de las pérdidas técnicas,
- e) eficiencia en el consumo,
- f) reducción del consumo doméstico,
- g) reducción del consumo en la iluminación de las calles,
- h) reducción de grandes consumidores y consumo irresponsable,
- i) fomento de las energías renovables,
- j) fomento de los vehículos eléctricos.

En España se ha desarrollado un proyecto 'Smart City' en el periodo 2009-2012 en la ciudad de Málaga.

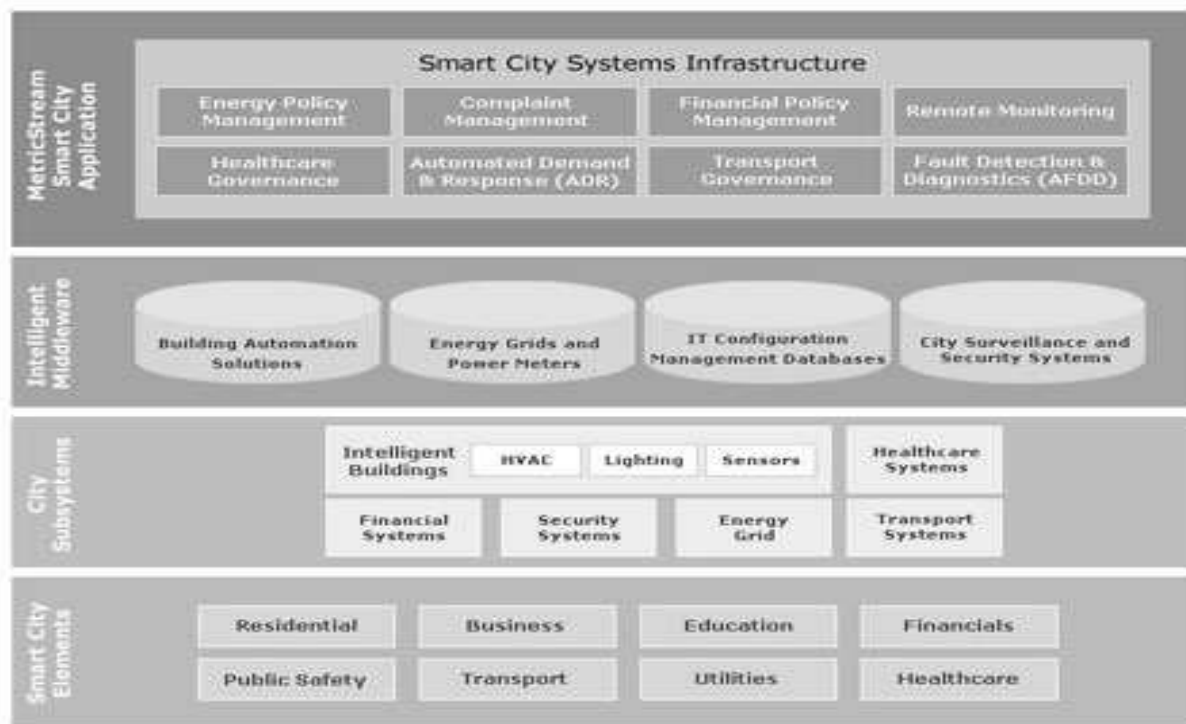


Figura 25 Ejemplo solución Smart City por MetricStream [23]

9.5 Edificios inteligentes

El término “edificios inteligentes” describe un conjunto de tecnologías utilizadas para hacer que su diseño, construcción y funcionamiento sean más eficientes, y que se aplica tanto a las construcciones existentes como a las nuevas. Este conjunto incluiría sistemas de gestión de edificios (SGE) que manejarían los sistemas de calefacción y refrigeración según las necesidades de los ocupantes o software que desconectaría todos los ordenadores personales y monitores después de que todo el mundo se haya ido a casa. Los datos también pueden utilizarse para identificar oportunidades adicionales para mejorar la eficiencia.

Las TIC ofrecen una mayor oportunidad para reducir las emisiones de este sector (un 15% en 2020). En concreto, un estudio de los edificios de Norteamérica indica que mejorar el diseño, la gestión y la automatización de los edificios puede reducir sus emisiones en un 15%.

En Madrid se celebrará el “I Congreso Edificios Inteligentes” los días 23 y 24 de Octubre de 2013.

9.6 Logística inteligente

La logística en el transporte (incluyendo envasado, transporte, almacenamiento, compra del consumidor y residuos) es inherentemente ineficiente.

La eficiencia energética del transporte de pasajeros y de mercancías está determinada por la composición de la flota (número y tipo de vehículos), la utilización de vehículos (las tasas de ocupación y los factores de carga) y características de conducción (velocidad, distancia).

En el caso del transporte de mercancías, el consumo de energía también depende de las características de las mercancías transportadas (por ejemplo, el peso y el volumen de las mercancías).

La logística inteligente comprende una gama de herramientas de software y hardware que controlan, optimizan y dirigen las operaciones, que ayudan a reducir la necesidad de almacenamiento para inventario, consumo de combustible (ecoconducción), optimización de rutas y frecuencia de viajes con los vehículos vacíos o parcialmente cargados, e-comercio, etc.

El transporte representa casi un tercio del consumo final de energía y alrededor de un cuarto de las emisiones totales de CO₂. Mejoras de eficiencia energética en el transporte pueden resultar en una reducción considerable en el consumo de energía y emisiones de CO₂. La optimización de la logística utilizando las TIC produciría una reducción del 16% en las emisiones de transporte y una reducción del 27% en las emisiones de almacenamiento.

10 Anexo II: detalles de implementación

Para la implementación del sistema se ha seleccionado un *framework* que evita la codificación de elementos presentes en cualquier aplicación web. Sobre este *framework* se ha desarrollado una serie de funcionalidades que complementan el *framework* y que servirían para cualquier desarrollo. Sobre lo anterior se ha desarrollado un conjunto de funcionalidades más específicas para el sistema, agrupadas en lo que llamamos librerías. Por último se ha implementado, con todo lo anterior, la herramienta que provee la funcionalidad requerida en unas agrupaciones de código que hemos llamado "*plugin*".

10.1 Selección del *framework*: Drupal (CMS)

"Un *framework* se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta"

Características básicas:

- Control de URL y sesiones.
- Acceso a datos.
- Autenticación y control de acceso.

Criterios de selección del *framework*:

- de **desarrollo web** puesto que es el entorno definido para el sistema.
- **código abierto** por ahorro de costes y filosofía.
- lenguaje de programación **PHP** por requerimientos.

Con los condicionantes anteriores hay tres opciones principales: Wordpress, Drupal o Joomla. La elección ha sido Drupal dado que parece el ideal para "sitios complejos, avanzados y versátiles, que requieren la organización de datos complejos". Al igual que en la elección del lenguaje de programación, el factor de más peso ha sido el conocimiento del framework por el desarrollador.

10.1.1 Drupal, módulos del core

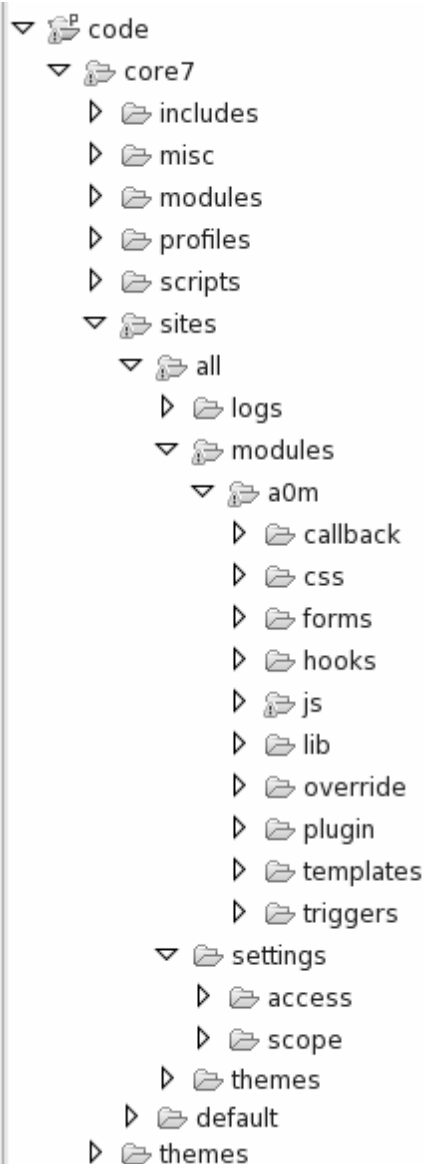
Se han seleccionado los módulos de core que se muestran en la siguiente tabla, aunque para la aplicación desarrollada únicamente son necesarios los módulos: **system, user, node, menu y block.**

Módulo	Función
Block	Controla los componentes básicos visuales de una página se construye con. Los bloques son cajas de contenido representado en una zona o región, de una página web.
Comment	Permite a los usuarios comentar y debatir el contenido publicado.
Field	API de campos para agregar campos a entidades como nodos y usuarios.
Field SQL	Almacenes de datos de campo en una base de datos SQL.
Field UI	Interfaz de usuario para la API de campo.
File	Define un tipo de campo de archivo.
Filter	Filtra el contenido en la preparación para la visualización.
Forum	Proporciona un foro de discusión.
Image	Proporciona herramientas de manipulación de imágenes.
List	Define la lista de tipos de campo. Usar con opciones para crear listas de selección.
Menu	Permite a los administradores personalizar el menú de navegación del sitio.

Node	Permite que el contenido que se presentará en el sitio y se muestra en las páginas.
Options	Define la selección, casilla de verificación y radio reproductores botón para el texto y los campos numéricos.
PHP filter	Permite incrustado código / snippets PHP para ser evaluado.
Poll	Permite su sitio para captar votos en diferentes temas, en forma de preguntas de opción múltiple.
Search	Permite la funcionalidad sitio-ancha búsqueda de palabras clave.
System	Maneja la configuración general del sitio para los administradores.
Taxonomy	Permite la categorización de contenido.
Text	Define los tipos de campos de texto simples.
Trigger	Permite a las acciones que se dispararon a ciertos eventos del sistema, como cuando se crea nuevo contenido.
User	Administra el registro del usuario y el sistema de inicio de sesión.

Tabla 3 Módulos del core de Drupal seleccionados

10.2 Estructura de ficheros



Bajo el directorio designado como [code], se encuentra una carpeta con el nombre **core7**, que contiene la versión 7 de drupal. En la ruta [code]/core7/sites/all, referenciada en adelante como [custom], se encuentran las siguientes carpetas:

- **logs:** ficheros de log, en concreto el log de errores generado desde la aplicación.
- **module/a0m:** contiene el código del módulo desarrollado (a0m).
- **settings:** ficheros de configuración adicionales

No se ha realizado modificación alguna en los ficheros de la distribución base, con lo que la actualización de versiones menores no debe constituir ningún problema. A la hora de realizar una actualización del código base de Drupal se volcará en la en la carpeta correspondiente (core7), y posteriormente se ejecutará el *script* de

actualización (update.php).

10.3 Funcionalidad transversal

Funcionalidades añadidas de forma transversal:

- **devel:** conjunto de funciones de ayuda al desarrollo.

- **multienvironment** - entorno local y de producción.
- **allinone**: desarrollo utilizado un único módulo (sin contar el *core*).
- **accessbylevel**: acceso a los nodos por nivel.
- **scope**: el primer nivel del camino de la URL define un entorno.
- **customblock**: los bloques se definen en función del *scope* y tema usado.
- **plugin**: funcionalidad que puede ser añadida a los diferentes *scopes*
- **profiling**: informes para la optimización del sistema

Cada una de estas funcionalidades se detallan a continuación.

10.3.1 **devel - Ayuda al desarrollo**

El desarrollo ha sido realizado usando el IDE eclipse, y a pesar de tener un excelente depurador (Xdebug, Zend Debugger) se presenta la necesidad de ver la consulta SQL exacta que se estaba realizando.

Al programar consultas que se construyen dinámicamente a través de PHP Data Objects (PDO), la sentencia SQL queda oculta al programador. Cuando existe algún error es difícil diagnosticar el problema si se puede ejecutar manualmente la consulta exacta.

Como Drupal no dispone de ninguna función que permita obtener la consulta final a partir del objeto, se ha programado la función `a0m_get_query_string` modificando en el código extraído de la página:

<http://drupal.stackexchange.com/questions/51333/print-the-query-which-is-built-using-db-select>

Una vez construido el objeto que contiene la definición y los parámetros de la misma, se pasa el objeto a la función `a0m_get_query_string`, que devolverá una cadena texto con la sentencia SQL que se puede ejecutar en una línea de comandos.

10.3.2 ***multienvironment*** - entorno local y de producción

En el desarrollo y/o mantenimiento de un sistema existe la necesidad de disponer al menos de dos entornos uno de pruebas/desarrollo y otro de producción.

Cualquier sistema configurable, puede serlo para trabajar en múltiples entornos, pero lo que se contempla aquí es que usando una única configuración, ésta nos permita la ejecución en varios entornos.

Este sistema implementa dos entornos: *local* y *producción*.

Para ello se define la constante: ***A0M_LOCAL_ENVIRONMENT*** que solo estará definida en el entorno local. Una o varias funciones (*_a0m_local_environment()* , *_a0m_prod_environment()*) nos indicarán en qué entorno estamos trabajando.

El conocimiento del entorno debe producirse en la fase más temprana posible, en este sistema se define en la carga inicial de la configuración común (*d0_settings.ini.php*), para ello se consulta que en el nombre del *host* aparezca la cadena 'local'.

Las funciones utilizadas para la consulta del entorno son:
_a0m_local_environment(): TRUE si el entorno es local.

_a0m_prod_environment(): TRUE si el entorno es producción.

Estas funciones devuelven cada una el resultado opuesto de la otra y aunque sean redundantes permiten que la lógica del código que las usa sea más claro.

10.3.3 ***allinone*** - Todo en un módulo

En un desarrollo convencional de drupal, se habría planificado una serie de módulos que nos aportarían la funcionalidad deseada. En este sistema toda la funcionalidad se ha desarrollado un único módulo.

▼ a0m	
▶ callback	En la raíz se encuentra el fichero principal (.module), el fichero de instalación (.install) y el ficheros de información de módulo (.info) requerido por drupal.
▶ css	
▶ forms	
▶ hooks	
▶ js	
▶ lib	En la carpeta callback se encuentran la implementación de las funciones usadas para construir determinadas salidas (*.page_callback.php). Las carpetas css y js contienen las
▶ override	
▶ plugin	
▶ templates	hojas de estilo y los ficheros javascripts usados. La carpeta

forms alberga las alteraciones a los fomularios del core que han sido modificados. La carpeta **hooks** almacena la implementación de los *hooks* usados, estos han sido divididos y almacenados a su vez en carpetas que hacen referencia al módulo que define el *hook*. La carpeta **lib** contiene tanto las librerías externas como las desarrolladas para este proyecto, las cuales se detallarán más adelante. En la carpeta **override** se encuentra el código que de una forma más directa altera el comportamiento "standard" del sistema. La carpeta **templates** contiene las plantillas definas para las funcionalidades transversales del módulo. La carpeta **plugin** contiene una serie de subcarpetas cada una de las cuales implementa una funcionalidad que sólo estará presente en aquellos "*scopes*" que declaren su uso (más adelante se detalla en profundidad los términos "*scope*" y "*plugin*").

10.3.4 *accessbylevel* - acceso a nodos por nivel

Drupal provee un sistema de control de acceso a los nodos bastante completo, aunque en algunos casos puede ser un sistema algo "pesado". Por ello se ha implementado un sistema bastante ligero que respetando el sistema standard de Drupal, también concede el acceso a los nodos comparando un nivel asignado a este nodo, nivel de acceso del nodo, y el mayor nivel resultante entre el concedido a un usuario determinado y el concedido a cualquiera de los roles que se haya asignado al usuario.

Por ejemplo, un nodo tiene un nivel de acceso de 7 el usuario tiene un nivel de acceso de 3 y tiene un roles con niveles de acceso de 0,2,6 y 9. Su nivel de acceso

será de 9 (el mayor entre 3 como usuario y 9 por el mayor nivel de acceso de los roles asignados) y como este es superior al nodo (nivel 7) el usuario tendrá permisos para visualizarlo.

Para la asignación de este nivel de acceso se ha tenido que alterar los formularios de creación/modificación de usuarios y los fomularios de creación/modificación de roles.

Los niveles de acceso (con valores de 0 a 255) se pueden establecer a través de la clave de configuración '**a0m_access_level**' y se ha establecido los siguientes: Mínimo(0), básico(1), avanzado(2), premium(3), lector(4), autor(5), editor(6), diseñador(7), desarrollador(8), administrador(9). Inicialmente todos los nodos, usuarios y roles tienen un nivel de 0, con la excepción del súper-usuario (uid=1) al que se le asigna el valor 255.

El nivel de acceso por rol se calcula en el momento que se determina que un usuario esta autenticado (session.X.inc)

La función que detemrina el nivel de acceso de un usuario es: `_a0m_access_level()`.

Esta funcionalidad obliga a introducir cambios en el esquema de Drupal. El campo **A0M_ACCESS_LEVEL** ha sido añadido a las entidades **role**, **users** y **node**.

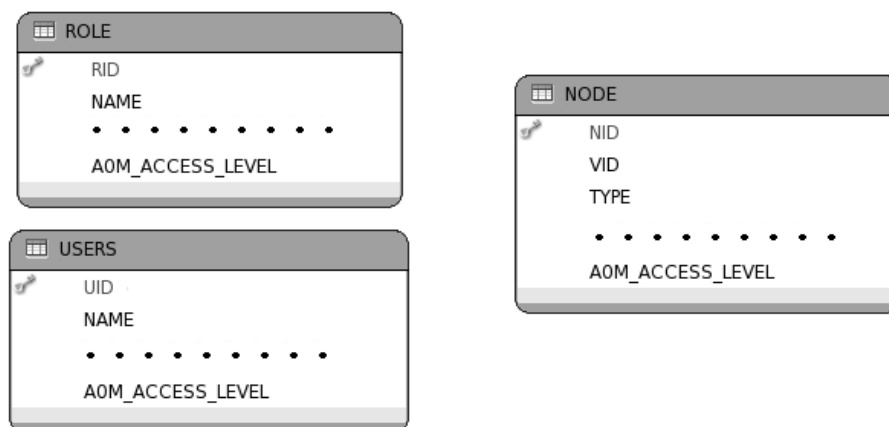


Figura 26 Cambios en el esquema para la funcionalidad *accessby/level*

Para realizar estos cambios en la estructura de las bases de datos se utiliza el mecanismo de actualizaciones que dispone drupal (ver el archivo: `[custom]/modules/a0m/a0m_install.7.inc.php`, función `a0m_update_7022()`).

Aunque este mecanismo ha sido introducido para permitir acceso a nodos se podría extender con facilidad a otras funcionalidades (acceso a servicios, *scopes*, *plugins*, etc.).

10.3.5 **scope - encapsulamiento de contenidos**

Es un mecanismo por el que se pretende:

- a) restringir el acceso a funcionalidades (bloques, *plugins*),
- b) definir la versión de drupal a usar para ese conjunto de funcionalidades,
- c) delimitar el acceso a los contenidos (nodos, comentarios, taxonomías).

Las acciones de administración se podrán realizar desde cualquiera de los *scopes* aunque se puede restringir el acceso a estas funciones.

Si el formato genérico una URL usada con HTTP es:

`http://maquina:puerto/camino/fichero`

el *scope* se determina a través del conjunto de caracteres que se encuentren antes de una barra (`/`) en el camino.

Ejemplos:

`http://www.host.es/lar/index.html => scope: lar`

`http://www.host.es/svcchp/chapter_tool/1/ => scope: svcchp`

Todos los *scopes* comparten las mismas configuraciones genéricas (comunes y de versión) y además poseen unas configuraciones adicionales que se encuentran en:

`[custom]/settings/access/<nombre_scope>.access.php'`

`[custom]/settings/scope/<nombre_scope>.settings.php`

El chequeo de acceso que se puede realizar en este punto es genérico, sin tener en cuenta al usuario ya que todavía no hemos validado la información de usuario.

No se ha llegado a automatizar la creación de los *scopes*, es por ello que para añadir un nuevo *scope* haya que realizar una serie de acciones manuales:

1. En [custom]/settings/d0_settings.ini.php, debe asignarse una identidad al *scope*, usando la clave de configuración `a0m_scopes`.

Por ejemplo:

```
$conf['a0m_scopes'] = array(..., 'svcchp' => array('id' => 10004), ...);
```

2. Se crea una tabla con la misma estructura que `menu_router` y con el prefijo del *scope*, y se especifica en el fichero de configuración `<nombre_scope>.settings.php`.

Por ejemplo:

```
$databases['default']['default']['prefix']['menu_router'] = 'svcchp_';
```

Esta acción no es obligatoria.

3. Si se ha especificado una tabla `menu_router` propia para el *scope*, debe ejecutarse el proceso de actualización invocando a la url:
`http://host/<nombre_scope>/update.php`

Se ha modificado el esquema de las tablas: **node**, **block**, **comment** y **taxonomy_term_data** para que cada *scope* solo tenga acceso a sus contenidos.

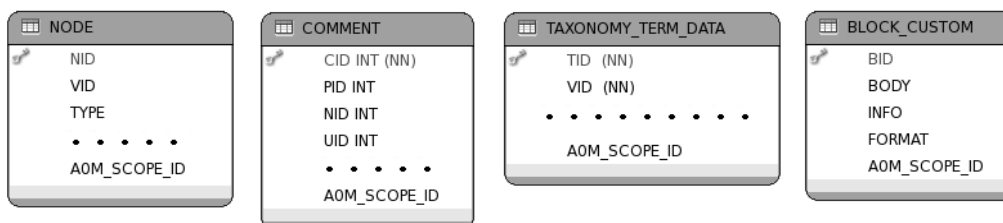


Figura 27 Cambios en el esquema para la funcionalidad *scope*

Para limitar el acceso a los contenidos hemos actuado sobre el *hook* **query_alter** añadiendo una reglas adicionales a la consultas para filtrar los contenidos por el *scope*.

Existe un *scope* de defecto que tiene acceso a todos los contenidos. Este *scope* está definido por las constantes *A0M_SCOPE_HOME* y *A0M_SCOPE_ID_HOME*. Estas constantes las podemos encontrar en el fichero:
[custom]/settings/d0_settings.ini.php

Los bloques definidos usando este *scope* pueden declararse como "universales", lo que permite ser utilizados por el resto de *scopes*.

Cada *scope* hace uso de unas caches independientes, para ello se ha extendido la clase que implementa las caches del sistema. La clase *a0m_DrupalDatabaseCache* debe ser declarada usando la variable de configuración 'cache_default_class'.

Por último, al llamarse a una configuración diferencia por cada *scope*, y realizarse esta llamada una vez cargada el resto de configuraciones, tenemos la posibilidad de sobrescribir cualquier variable de configuración lo que se traduce en la posibilidad de definir diferentes temas para cada *scope*, etc.

10.3.6 **customblock - bloques por scope y tema**

Para dar más flexibilidad al sistema de bloques se ha desarrollado varias funcionalidades que afectan a la definición, configuración y selección del bloque.

Para la definición:

- se puede especificar en qué *scopes* es visible el bloque

En cuanto a la configuración:

- un bloque puede ser instanciado con diferentes configuraciones.
- las configuraciones pueden ser incluidas programáticamente.

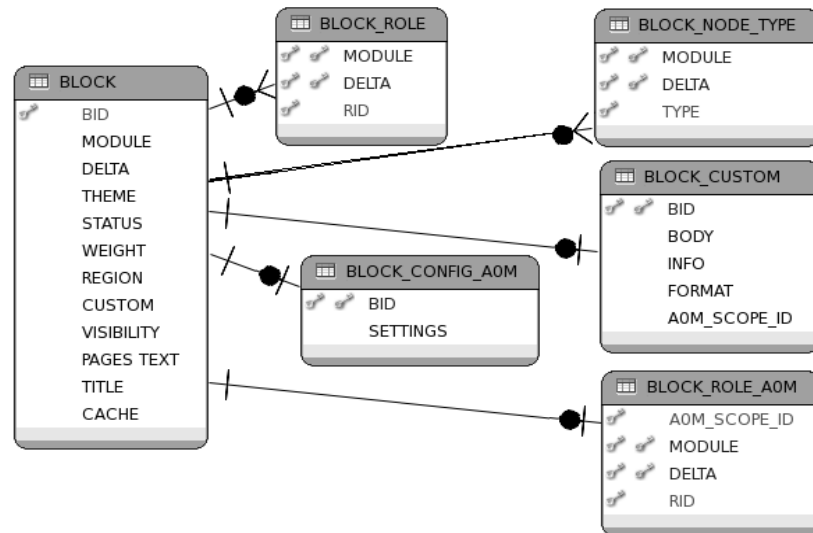


Figura 28 Esquema de datos para la funcionalidad *themeblock*

A la hora de la selección:

Los bloques son seleccionados en función del tema que esté seleccionado. Esto nos permite, por ejemplo, incluir bloques diferentes en las páginas de administración si utilizamos para estas páginas un tema diferenciado.

10.3.7 *plugin* - funcionalidad encapsulada y opcional

El concepto de *plugin* es el mismo que el de los módulos, con la diferencia de que estos *plugins* no deben ser instalados, su mera presencia es suficiente y el ámbito de carga es el *scope*.

Un *plugin* debe ser declarado en la configuración de todos aquellos *scopes* que quieran usarlos a través de la variable de configuración 'a0m_scope_plugins' en el fichero <nombre_scope>.setting.php.

Por ejemplo:

```
$conf['a0m_scope_plugins'] = array('xid_client', 'chapter_client');
```

Los *plugins* aportan a los *scopes* toda la funcionalidad no común desarrollada en el módulo único a0m. Al cargarse únicamente en aquellos *scopes* en los que se declare su uso, la cantidad de código, memoria etc usada por cada *scope* se restringe

verdaderamente a la funcionalidad que se pretende aportar, ejecutando sólo aquellas inicializaciones necesarias.

Los *plugins* se localizan en la carpeta [custom]/modules/a0m/plugin/. Cada *plugin* se implementa dentro de una carpeta que tiene obligatoriamente el mismo nombre del *plugin*. Ejemplos de *plugin* implementados: xid, loc, chapter, etc.

Se ha implementado un conjunto de funciones para que el formato de estos *plugins* se mantenga tan uniforme como sea posible, definiéndose unas reglas implícitas para el emplazamiento de cada tipo de fichero, aunque también permite no seguir las mismas para más flexibilidad (plantillas, ficheros css, ficheros js, etc). El fichero base de un *plugin* tiene la extensión: **.a0mplugin**.

10.3.8 *profiling* - diagnóstico y optimización del sistema

Para optimizar rendimientos y diagnosticar problemas relativos a los mismos, necesitamos tener fundamentalmente medidas de consumos de tiempo y memoria.

Una de las prioridades de este sistema es que sea lo menos intrusivo y lo más ligero posible durante el tiempo de ejecución.

Modificando el código del módulo ***profiling***, se ha introducido un parámetro opcional (*profiling*) que puede ser introducido en cada página y que nos reporta datos de consumo de tiempos y memoria. Estos datos son añadidos como un comentario HTML al final de la página.

El código deja las bases puestas para un desarrollo más amplio de estos informes y está preparado para volcar los datos en la base de datos para análisis más complejos.

Por ejemplo: una llamada a la página de la herramienta de contadores con el *profiling* activado se realizaría a través de la siguiente url:

`http://<host>/svccdef/counterdef_tool?profiling=1`

y al final del código fuente encontraríamos los datos:


```
<!--  
path: counterdef_tool  
created: 1371276625  
query_count: 12  
duration: 0.525256872177  
memory: 24117248  
...  
-->
```

10.4 Librerías

Bajo esta denominación se ha agrupado una serie de clases y funciones usadas por uno o más *plugins*.

La localización del código es:

- para el código php: [custom]/modules/a0m/lib/<libreria>
- para el código js: [custom]/modules/a0m/js/lib/<libreria>
- para el código css: [custom]/modules/a0m/css/lib/<libreria>

Este código hace uso de las funciones provistas por el *framework*, esto implica que no pueden ser utilizadas, sin realizar modificaciones, fuera del mismo.

Las librerías desarrolladas son:

- a) JDT - jQuery Datatables: Manipulación de tablas
- b) JTR - jQuery jTree: Manejo de estructuras jerárquicas
- c) RPC_XML_a0m - Interfaz xmlrpc y envoltura pseudoREST
- d) OWN - Repositorio común de objetos
- e) PHP_CLIENT - Interfaz php con servicios

En los siguientes apartados se comentan más extensamente cada una de ellas.

10.4.1 JDT - jQuery Datatables

Para el manejo de los datos de las tablas utilizados por el sistema se podía haber implementado una serie de paneles de administración al uso en el *framework* drupal, pero se ha preferido implementar un sistema CRUD (*Create/crear*, *Retrieve/obtener*, *Update/actualizar* y *Delete/borrar*) más ágil y que permitiese la actualización simultánea de más de una tabla en una misma página.

Para elegir este sistema se ha seleccionado una librería javascript para el código del lado cliente. Han sido analizadas:

- extJs (<http://www.sencha.com/products/extjs/>)
- jQuery EasyUI (www.jeasyui.com)
- jTable (<http://www.jtable.org/RealTime>)
- jqxGrid (<http://jquerygrid.net/>)
- Datatables + jquery plugins (<http://www.datatables.net/>)

La librería elegida fue Datatables, ponderando principalmente el uso de la librería jQuery y su documentación. Tanto su descripción como la API y los ejemplos encontrados son de gran calidad.

La implementación no es la más trivial ya que para completar la funcionalidad hay que hacer uso de un total de nueve plugins jQuery, que detallamos más adelante.

La librería la componen los siguientes elementos:

Parte **Javascript**: en `[custom]/modules/a0m/js/lib/JDT`, que sirve para iniciar los diferentes plugins.

Parte **CSS**: en `[custom]/modules/a0m/css/lib/JDT`, reglas css para mostrar las tablas.

Parte **PHP**: en [custom]/modules/a0m/lib/JDT, que aportan del lado servidor lo siguiente:

- plantilla para generar el html necesario (*datatables_table.tpl.php*) para una tabla.
- servidor de datos que interpreta las peticiones ajax del cliente. Ejecuta las acciones CRUD necesarias y devuelve un conjunto de datos en formato JSON o una indicación de resultado de la ejecución (con éxito o error)
- inclusión en la página de los archivos css necesarios.
- inclusión en la página de las llamadas a archivos javascript necesarios.
- conversión de los datos de la tabla en estructuras JSON para la inicialización de los diferentes *plugins* del lado del cliente.
- creación del formulario necesario para la inserción de registros.
- creación de un bloque drupal partiendo de los datos de configuración

10.4.1.1 Plugin jQuery Datatables (v1.9.4)

URL: <http://www.datatables.net/>

Datatables es un *plugin* muy flexible, basado en los fundamentos de la mejora progresiva, lo que añade **controles avanzados** de interacción a cualquier tabla HTML. Entre sus características principales se encuentran:

La paginación de longitud variable, filtrado de elementos "*on-the-fly*", posibilidad de ordenación de varias columnas (con detección de tipo de datos), manejo inteligente del ancho de las columnas, obtención de datos a través de múltiples fuentes (DOM , array Javascript , Ajax), **procesamiento del lado del servidor** (PHP, C #, Perl, etc) , y totalmente **internacionalizable**.

10.4.1.2 Plugin jQuery jeditable (v.1.7.2)

URL: <http://www.appelsiini.net/projects/jeditable>

Permiten hacer clic y editar el contenido de los diferentes elementos HTML.

Su funcionamiento es simple: cuando el usuario hace clic en el texto de la página web el bloque de texto se convierte en un formulario, el usuario edita contenidos y presiona el botón enviar, el formulario se envía al servidor web y se guarda.

10.4.1.3 Plugin jquery-ui (v1.9.2)

URL: <http://jqueryui.com/>

jQuery UI es un conjunto de interacciones de interfaz de usuario, efectos, *widgets* y temas construidos sobre la biblioteca jQuery.

En este desarrollo hemos usado el tema "*smoothness*".

10.4.1.4 Plugin jQuery validation (v1.11.0)

URL: <http://jqueryvalidation.org/>

Este *plugin* facilita la validación de formularios del lado cliente y ofrece numerosas opciones de personalización. Ofrece métodos de validación para numerosos campos (URL, correo electrónico, etc) y proporciona una API para escribir métodos para nuevos campos. Los mensajes de error se muestran por defecto en inglés pero existe traducción para el español.

10.4.1.5 Plugin jQuery dataTables.editable (v2.3.3)

URL: <http://code.google.com/p/jquery-datatables-editable/>

Este *plugin* agrega las **funcionalidades CRUD** de gestión de datos:

- añadir nuevos registros a la tabla
- selección y eliminación de registros

- edición de los campos en línea

en el lado del cliente.

Sólo se requiere un código de servidor que intérprete y ejecute solicitudes.

En nuestro sistema el código servidor se ha implementado en PHP.

10.4.1.6 Plugin jQuery dataTables.columnFilter (v1.5.0)

URL: <https://code.google.com/p/jquery-datatables-column-filter/>

Este *plugin* permite **añadir filtros a cada columna** individualmente. El filtrado puede realizarse a través de un campo texto o de un campo combo.

10.4.1.7 Plugin jQuery KeyTable (v1.1.7)

URL: <http://www.sprymedia.co.uk/software/KeyTable/>

Keytable **permite navegar por una tabla HTML usando el teclado**. La celda bajo el foco se muestra a través de la clase CSS (*'focus'*). A través de su API podemos asignar (para eventos como *'focus'*, *'blur'*, *'esc'*, etc) funciones manejadoras de eventos.

10.4.1.8 Plugin jQuery alerts (v1.1)

URL: <https://github.com/aurels/jquery.alerts>

Sirve para **mostrar al usuario**, en *pop-ups*, **mensajes** o peticiones de confirmación para las acciones.

10.4.1.9 Plugin jQuery jloadable (toolkit)

Creado para completar la funcionalidad. Permite definir los valores de los campos de un formulario con datos obtenidos del servidor en el momento de ser mostrado.

10.4.2 JTR - jQuery jsTree

Tanto para elaborar una estructura jerarquizada de localizaciones como para estructurar una clasificación de contadores, se tiene que lidiar con **estructuras en árbol** para las cuales no es sencillo construir una interfaz de usuario amigable. Es por ello por lo que se ha seleccionado una librería ya construida que simplificase el trabajo y ahorrarse tiempo de desarrollo.

Al igual que para el sistema CRUD, se ha estudiado diferentes librerías cliente en javascript que manejan estructuras en árbol:

- treeview (<http://jquery.bassistance.de/treeview/demo/>)
- jsTree (<http://www.jstree.com/>)
- Dynatree (<http://code.google.com/p/dynatree/>)

Para el sistema se ha seleccionado jsTree. Es un **plugin jQuery**, con una documentación notable y una API bien construida. Entre las funcionalidades más destacables está el uso de múltiples fuentes de datos (HTML, JSON, XML), la **carga dinámica de datos a través de llamadas AJAX**, el uso de "**drag & drop**" para el movimiento de nodos, su facilidad de configuración, el soporte de temas, la posibilidad de navegación a través de teclado y el **soporte multilinguaje**.

El grueso del desarrollo ha consistido en crear un conjunto de funciones que simplifican la creación del html necesario para el uso de la librería cliente y fundamentalmente el diseño de una estructura genérica de datos y la creación de los procedimientos necesarios para interpretar las peticiones ajax del cliente y proceder a la manipulación correcta de los datos.

Se ha diseñado la estructura para que esta fuese capaz de **albergar más de un árbol** a la vez. Una característica necesaria era que los nodos tuviesen una **jerarquía** y esta se respetase a la hora de construir el árbol, lo cual también ha sido tomando en cuenta a la hora de la implementación.

La librería la componen los siguientes elementos:

Parte **Javascript**: en [custom]/modules/a0m/js/lib/JTR, que sirve para iniciar el *plugin* jsTree con los datos JSON incrustados en el html de la página.

Parte **CSS**: en [custom]/modules/a0m/css/lib/JTR, reglas css para mostrar los bloques de árboles.

Parte **PHP**: en [custom]/modules/a0m/lib/JTR, que aportan del lado servidor lo siguiente:

- plantilla para generar el html necesario (*JTR_jstree.tpl.php*) para visualizar un árbol.
- servidor de datos JSON que interpreta las peticiones ajax del cliente y ejecuta diversas acciones (reconstruir, chequear, analizar) sobre el árbol (*JTR_json_jstree_server()*).
- inclusión en la página de los archivos css necesarios (*JTR_jstree_css()*).
- inclusión en la página de las llamadas a archivos javascript necesarios e inclusión de datos de inicialización en formato JSON para el cliente (*JTR_jstree_js()*).
- Creación de un bloque drupal que visualice el árbol (*JTR_jstree_drupal_block()*).
- Una clase (ver fichero *JTR_jstree.class.inc*) que contiene los métodos necesarios para la manipulación del árbol: *analyze*, *reconstruct*, *check*, *move_node*, *remove_node*, etc.

10.4.3 RPC_XML_a0m - Interfaz xmlrpc

El *framework drupal* soporta de forma **nativa** el protocolo **XMLRPC**, lo que se ha desarrollado es un conjunto de funciones que se ha utilizado en cada uno de los servicios y que están relacionados con su uso.

Para la parte servidora (archivo *RPC_XML_a0m_server.inc*):

- estructura del mensaje de respuesta
- construcción del mensaje de error

Para dar mayor flexibilidad, y cumplir con las **especificaciones**, se ha desarrollado una capa pseudo **REST** que juega con el nombrado de las URL y los parámetros de la petición convirtiendo internamente estas peticiones en llamadas XMLRPC y devolviendo los datos en el formato indicado en la petición.

Para la parte cliente (archivo `RPC_XML_a0m_client.inc`):

- declaración de las urls que serán atendidas (`RPC_XML_a0m_client_menu_items()`)
- conversión de la respuesta xml en el formato adecuado (`RPC_XML_a0m_client_xmlrpc()`).

Por último, se ha desarrollado un sistema de páginas que sirven para probar inicialmente los métodos, al que se ha llamado cliente web.

Para la parte cliente web (archivo `RPC_XML_a0m_client_web.inc`):

- declaración de las urls necesarias (`RPC_XML_a0m_client_web_menu_items()`)
- generación y proceso de los formularios web (`RPC_XML_a0m_client_web()`).

10.4.4 OWN - Repositorio común de objetos

Se ha implementado una clase (OWN) que sigue el patrón de diseño *singleton* y que es instanciada al inicio de la ejecución de cada página. Está concebida para ser el **almacén de objetos** de otras clases **con patrón singleton**. Para ello se ha desarrollado otra clase (OWN_SINGLETON) de la que deberán extender aquellas clases que quieran usar este sistema (la clase PHP_CLIENT que se trata en el siguiente punto es una de ellas).

10.4.5 CLIPHP - Interfaz php con servicios

En el afán de ejecutar la misma acción independientemente del conducto por el que llegue la orden, se ha desarrollado un mecanismo que permite definir una serie de métodos que finalmente llamarán al método xmlrpc adecuado.

En el archivo `[custom]/modules/a0m/plugin/counterdef_client/counterdef_client.a0mplugin` se puede ver como se declaran una serie de métodos, para cada uno de los cuales se indica:

- URL a la que se atacará.
- tipo de petición (POST, GET, PUT)
- parámetros necesarios y opcionales
- el campo o campos de retorno

Un ejemplo del uso de estos métodos lo encontramos en el archivo `[custom]/modules/a0m/plugin/counterdef_tool/_counterdef_tool.inc`:

```
$oCLI = CLIPHP::getSharedObject();
```

```
$result = $oCLI->counterdef_add_class($class_code, $class_name);
```

10.5 Funcionalidad específica

Se han diseñado los diferentes *plugins* que implementan cada uno de los servicios ofrecidos por los subsistemas y se ha creado un "scope" por cada subsistema, asignando a cada *scope* los *plugins* necesarios.

10.5.1 Plugins

Para cada sistema se ha desarrollado tres *plugins*, uno que implementa cada uno de los métodos XMLRPC necesarios, otro que implementa el interfaz cliente PHP para acceder a los métodos y un último *plugin* que implementa las herramienta cliente de manipulación de tablas y árboles.

El listado completo de *plugins* es:

xid: parte servidora y cliente web para el servicio identidad única.

xid_client: definición del interfaz PHP para el servicio identidad única

xid_tool: GUI para identidad única.

spot: parte servidora y cliente web para las localizaciones.

spot_client: definición del interfaz PHP para las localizaciones.

spot_tool: GUI para localizaciones.

forest: parte servidora y cliente web para el manejo de árboles.

forest_client: definición del interfaz PHP para el manejo de árboles.

forest_tool: GUI para el manejo de los árboles.

chapter: parte servidora y cliente web para el manejo de clasificaciones.

chapter_client: definición del interfaz PHP para el manejo de clasificaciones.

chapter_tool: GUI para el manejo de clasificaciones.

counterdef: parte servidora y cliente web para el manejo de indicadores.

counterdef_client: definición del interfaz PHP para el manejo de indicadores.

counterdef_tool: GUI para el manejo de indicadores.

10.5.2 Scopes - subsistemas

Para implementar cada subsistema hemos creado un *scope* consiguiendo con esto un aislamiento máximo de cada una de las funciones.

Los *scopes* se declaran en el archivo [custom]/settings/d0_settings.ini.php, usando la variable de configuración 'a0m_scopes'.

La asignación de los *plugins* a cada uno de los *scopes* (subsistemas), hay que editar sus respectivas configuraciones, que se encuentran en el directorio [custom]/settings/scope en el archivo <scope>.settings.php.

La asignación de los *plugins* a cada uno de los subsistemas (*scopes*) se detalla a continuación:

Subsistema Identidad Única (SIU): 'xid'

Subsistema Localizaciones (SLO): 'spot', 'xid_client', 'spot_tool'

Subsistema Arbol Jerárquico (SAJ): 'forest', 'xid_client', 'forest_tool', 'spot_client', 'forest_client'

Subsistema Clasificación (SCL): 'chapter', 'chapter_client', 'chapter_tool', 'xid_client'

Subsistema de Gestión de Indicadores (SGI): 'counterdef', 'counterdef_client', 'counterdef_tool', 'xid_client', 'chapter_client', 'spot_client'

10.5.3 Detalle de los interfaces para el Servicio svcid

Interfaz XMLRPC/pseudoREST

Operaciones:

- Obtener identidad única

URL: `http://localhost/svcid/xid/client/xmlrpc/identifier/get`

Parámetros:

`field_type` (alfanumérico) (por defecto: "---")

Respuesta: campo "xid" (XID)

- Obtener última identidad concedida de un grupo

URL: `http://localhost/svcid/xid/client/xmlrpc/identifier/last`

Parámetros:

`filter_field_type (XID_TYPE) (alfanumérico)`

Respuesta: campo "last" (XID)

- Añadir grupo de identidades

URL: `http://localhost/svcid/xid/client/xmlrpc/identifier/insert`

Parámetros:

`field_type (XID_TYPE) (alfanumérico)`

`field_xid (XID) (numérico) (opcional)`

Respuesta: Resultado de la operación.

- Modificar último id de un grupo de identidades

URL: `http://localhost/svcid/xid/client/xmlrpc/identifier/update`

Parámetros:

`filter_field_type (XID_TYPE) (alfanumérico)`

`field_xid (XID) (numérico)`

Respuesta: Resultado de la operación.

- Eliminar un grupo de identidades

Parámetros:

`filter_field_type (XID_TYPE) (alfanumérico)`

URL:

`http://localhost/svcid/xid/client/xmlrpc/identifier/delete/[json|php|text]`

Respuesta: Resultado de la operación.

Librería PHP

Operaciones:

- Obtener identidad única

Método PHP: `xid_get`

Parámetros:

`field_type (XID_TYPE) (alfanumérico)`

Respuesta: campo "xid" (XID) (numérico)

- Añadir grupo de identidades

Método PHP: `xid_add_type`

Parámetros:

`field_type` (XID_TYPE) (alfanumérico)

`field_xid` (XID) (numérico) (opcional)

Respuesta: Resultado de la operación.

- Modificar último id de un grupo de identidades

Método PHP: `xid_upd_type`

Parámetros:

`filter_field_type` (XID_TYPE) (alfanumérico)

`field_xid` (XID) (numérico)

Respuesta: Resultado de la operación.

- Eliminar un grupo de identidades

Método PHP: `xid_del_type`

Parámetros:

`filter_field_type` (XID_TYPE) (alfanumérico)

Respuesta: Resultado de la operación.

Referencias

- [1] ANAE. *Asociación de consumidores de energía* [en línea]. [Consultado 2 de agosto de 2013]. Disponible en: <http://www.asociacion-anae.org/noticias/la-nueva-directiva-europea-sobre-eficiencia-energetica-nos-afectara-mas-todos>
- [2] ACEE Consultores. *Sistemas de Monitoreo de Indicadores Energéticos* [en línea]. [Consultado 2 de abril de 2013]. Disponible en: <http://www.acee.com.mx/index.php/sistemas-de-monitoreo-de-indicadores-energeticos>
- [3] Gobierno de Australia. *Energy Savings Measurement Guide* [en línea]. [Consultado 2 de agosto de 2013]. Disponible en <http://energyefficiencyopportunities.gov.au/guides/energy-savings-measurement/>
- [4] sh0rtlived. *Smart meter, the new fatal weapon* [en línea]. [Colsultado 4 de agosto de 2013]. Disponible en: <http://sh0rtlived.wordpress.com/2011/11/10/smart-meter-the-new-fatal-weapon/>
- [5] ESMIG. *the 20-20-20 goals* [en línea]. [Consultado 4 de agosto de 2013]. Disponible en: <http://www.esmig.eu/smart-metering/the%2020-20-20%20goals>
- [6] Comisión Europea. *Mandato M/441* [en línea] [Consultado el 8 de abril de 2013]. Disponible en: <http://www.cen.eu/cen/Sectors/Sectors/Measurement/Documents/M441.pdf>
- [7] ENDESA. *Smart Metering* [en línea]. [Consultado el 24 de mayo de 2013]. Disponible en: <http://www.endesa.com/en/aboutEndesa/businessLines/principalesproyectos/Telegestion>
- [8] Meters & More Association. *Open communication technology* [en línea]. [Consultado el 24 de mayo de 2013]. Disponible en: <http://www.metersandmore.com/>
- [9] Yellow Strom. *Smart Meetering* [en línea]. [Consultado el 25 de mayo de 2013]. Disponible en: <http://qigaom.com/2009/07/02/the-worlds-coolest-utility-yello-stroms-got-smart-meters-that-tweet/>
- [10] Google. *PowerMeter Project* [en línea]. [Consultado el 25 de mayo de 2013]. Disponible en: <http://www.google.com/powermeter/about/>
- [11] SIEMENS. *New energy indicator helps operate plants and rooms energy-efficiently* [en línea]. [Consultado el 19 de junio de 2013]. Disponible en: <http://www.siemens.com/press/en/pressrelease/?press=/en/pressrelease/2012/infrastructure-cities/building-technologies/icbt201207035.htm>
- [12] Joe Celko. *Trees and hierarchies in SQL for smarties*. 2ª ed. Morgan-Kaufmann, 2012. ISBN 978-0-12-387733-8

[13] United Nations. *Energy Statistics Database* [en línea]. [Consultado el 4 de abril de 2013]. Disponible en: <http://data.un.org/Explorer.aspx?d=EDATA>

[14] Instituto Nacional de Estadística. *Energy Consumption* [en línea]. [Consultado el 4 de abril de 2013]. Disponible en: <http://www.ine.es/jaxi/menu.do?type=pcaxis&path=%2Ft04%2Fp01&file=inebase&L=1>

[15] Reegle Data - *Clean Energy Datasets* [en línea]. [Consultado el 10 de abril de 2013]. Disponible en: <http://data.reegle.info/developers/guide>

[16] Kybele. *Especificación de Requisitos según el estándar de IEEE 830* [en línea]. [Consultado el 21 de julio de 2013]. Disponible en: http://www.kybele.etsii.urjc.es/docencia/AIR_GIS_M/2012-2013/Material/ieee830.pdf

[17] Wikipedia. [en línea]. [Consultado el 11 de agosto de 2013]. Disponible en: <http://es.wikipedia.org/wiki>

[18] Gelmi Domínguez. *Interfaces De Entrada Y Salida* [en línea]. [Consultado el 16 de agosto de 2013]. Disponible en: <http://www.slideshare.net/BigbossH/interfaces-de-entrada-y-salida>

[19] SNITT . *GUIA SIEN METODOLOGIA DE INDICADORES* [en línea]. [Consultado el 16 de agosto de 2013]. Disponible en: http://www.snitt.org.mx/pdfs/bioenergeticos/Guia_SIEN_Metodologia_Indicadores.pdf

[20] Clúster E2TIC. *Eficiencia Energética y TIC* [en línea]. [Consultado el 23 de agosto de 2013]. Disponible en: http://www.rcysostenibilidad.telefonica.com/rconversa/assets/docs/debates/media/Eficiencia_Energetica_y_TIC.pdf

[21] Ministerio de Agricultura, Alimentación y Medio Ambiente. *BANCO PÚBLICO DE INDICADORES AMBIENTALES* [en línea]. [Consultado el 5 de julio de 2013]. Disponible en: http://www.magrama.gob.es/es/calidad-y-evaluacion-ambiental/temas/informacion-ambiental-indicadores-ambientales/BPIA_2012_Ficha_web_Energia_Renovables_tcm7-164641.pdf

[22] Ibáñez, P. *Ventas en España de coches eléctricos* [en línea]. [Consultado el 7 de julio de 2013]. Disponible en: <http://www.motorpasionfuturo.com/coches-electricos/ventas-en-espana-de-coches-electricos-en-2012>

[23] MetricStream. *Smart Cities Solutions* [en línea]. [Consultado el 7 de julio de 2013]. Disponible en: http://www.metricstream.com/solutions/smart_cities.htm